

Technická Univerzita v Liberci

Fakulta Textilní

DIPLOMOVÁ PRÁCE

Liberec 2008

Jiří Kula

TECHNICKÁ UNIVERZITA V LIBERCI
FAKULTA TEXTILNÍ

Studijní program: M3106 – Textilní inženýrství

Studijní obor: Netkané textilie

Katedra netkaných textilií

**STUDIUM HLADINOVÉHO ELEKTROSTATICKÉHO
ZVLÁKŇOVÁNÍ**
**ANALYSIS OF NEEDLE-LESS ELECTROSPINNING FROM
FREE SURFACE OF LIQUID**

Autor: Jiří Kula

Vedoucí diplomové práce: prof. RNDr. David Lukáš, CSc.

Konzultant diplomové práce: doc. RNDr. Aleš Linka, CSc.

Počet stran textu: 65

Počet obrázků: 43

Počet příloh: 2

TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta Textilní

Katedra netkaných textilií

Akademický rok: 2007/2008

ZADÁNÍ DIPLOMOVÉ PRÁCE

Jméno a příjmení: Jiří Kula

Studijní program: M3106 Textilní inženýrství

Studijní obor: Netkané textilie

Název tématu: **Studium hladinového elektrostatického zvlákňování**

Zásady pro vypracování:

- 1) Vypracujte literární rešerši na téma hladinového zvlákňování. Vypracujte literární rešerši na téma digitálního zpracování obrazu.
- 2) Provádějte experimenty s lištovými zařízeními pro elektrostatické zvlákňování s cílem získat experimentální data o vzájemném vztahu intenzity pole a počtu spontánně vytvořených kapalinových trysek.
- 3) Experimenty zaznamenávejte pomocí digitální kamery a zároveň sledujte napětí mezi elektrodami.
- 4) Získaná data zpracujte metodami analýzy obrazu a statistickými metodami.
- 5) Prozkoumejte možnosti řízení a analýzy experimentu s využitím Matlabu.

Abstrakt

Tato diplomová práce se zabývá studiem samoorganizace trysek na volné hladině polymeru během elektrostatického zvlákňování. Zvolený přístup využívá metod analýzy digitálního obrazu k odhalení vztahu mezi rozestupy trysek na jednorozměrné hladině kapaliny a napětím mezi elektrodami. Volná hladina polymeru je formována pomocí laboratorního přípravku, tzv. lištového spinneru, do úzké štěrbiny na spodní elektrodě. Díky tomuto řešení došlo k podstatnému zjednodušení experimentu, které dovoluje jasnější popis fyzikálních jevů na hladině. Dosažené výsledky by mohly posloužit jako výchozí bod pro rozšíření obdobného přístupu na sledování dvourozměrné hladiny, popřípadě ke konstrukci řídicího systému pro výrobu nanovláken např. na strojích typu NanospiderTM.

Klíčová slova

elektrostatické zvlákňování, bez-jehlové zvlákňování, obrazová analýza

Abstract

The thesis aims to study of self-organized jets during electrospinning from free surface of a liquid. Chosen way uses methods of digital image processing to discover the relationship between mutual distances within jets and electrostatic field being applied. Free surface of polymer solution is kept in linear shape by its enclosure in narrow clef. This special device restricts movement of surface of liquid into only one dimension along the clef, thus simplifies description of electrospinning phenomenon. Achievements made in this piece could be threatened as essential approach to study 2-dimensional liquid surfaces, or as a fundamental for further development of control system related to present NanospiderTM technology.

Keywords

needle-less electrospinning, digital image processing

Prohlášení

Prohlašuji, že předložená *diplomová* práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem v práci neporušil autorská práva (ve smyslu zákona č. 121/2000 Sb. O právu autorském a o právech souvisejících s právem autorským).

Souhlasím s umístěním *diplomové* práce v Univerzitní knihovně TUL.

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č.121/2000 Sb. o právu autorském, zejména § 60 (školní dílo).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé diplomové práce a prohlašuji, že **s o u h l a s í m** s případným užitím mé diplomové práce (prodej, zapůjčení apod.).

Jsem si vědom toho, že užít své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

V Liberci, dne 18. března 2008

.....

Podpis

Poděkování

Rád bych poděkoval vedoucímu práce Prof. RNDr. Davidu Lukášovi, CSc. za výbornou motivaci a odborné rozpravy nad řešeným tématem, dále Doc. RNDr. Alešovi Linkovi, CSc. a Ing. Marošovi Tunákovi za pomoc při návrhu algoritmů získání a zpracování dat. Technikům Ing. Pavlovi Pokornému a Filipu Sanetníkovi vděčím za realizaci technických řešení.

Použité značky a symboly

T	perioda
k	vlnový vektor
A, C_1, C_2	amplituda
y	poloha, výchylka
t	čas
m	hmotnost
ω	úhlová frekvence
φ	počáteční fáze
ρ	měrná hmotnost
γ	povrchové napětí
i	imaginární jednotka; platí: $i^2 = -1$
D	diskriminant
E_0	intenzita elektrického pole
ε	permitivita prostředí
px	z angl. pixel (obrazový element)
bit	z angl. binary digit (dvojková číslice); základní a nejmenší jednotka informace
$double$	z angl. double precision; formát ukládání číselné hodnoty, délka 64 bitů; 1. bit znaménkový, 11 bitů exponent, 52 bitů mantisa;
$uint8$	z angl. unsigned integer; celočíselná hodnota délky 8 bitů bez znaménka; platná ve dvojkové soustavě, odpovídající hodnoty v desítkové soustavě jsou 0-255
AVI	z angl. audio video interleaving, datový kontejner uvedený fy Microsoft, 1992; dovoluje kódování audia a videa do jediného souboru, dovoluje tak jejich jednoduché synchronní přehrávání
CRT	z angl. cathode ray tube; označení pro klasickou televizní obrazovku

<i>IEEE 1394</i>	standardní sériové rozhraní určené k rychlé komunikaci mezi PC a periferiemi
L_E, L_4, L_8	vzdálenost Euklidovská, v městských blocích, na šachovnici
<i>miniDV</i>	formát běžných videokamer; rozlišení 720x576px, 25 snímků za vteřinu
<i>PAL</i>	z angl. phase alternation line; standard kódování barevného televizního vysílání, používaný v Evropě (kromě Francie)
<i>prokládání (interlacing)</i>	způsob komprese obrazu v televizním vysílání; zpětná konverze na tzv. úplné snímky (<i>progressive</i>) se nazývá <i>deinterlacing</i>
<i>WAV</i>	z angl. Waveform audio format; zvukový formát vyvinutý fy IBM a Microsoft; většinou obsahuje nekomprimovaná data v pulzní kódové modulaci <i>PCM</i>
<i>WDM</i>	z angl. Windows Driver Model; jedná se o uniformní a obecný předpis pro vývoj ovladačů hardware pro OS Windows
<i>WMV9</i>	z angl. Windows Media Video 9; v současnosti jeden z nejlepších kompresních a dekompresních ztrátových algoritmů pro uchování digitálního obrazu, výhodou je vysoká rozšířenost díky implementaci ve Windows verze XP a vyšších

Seznam Toolboxů a pojmů souvisejících programováním v Matlabu

<i>Data Acquisition Toolbox</i>	sada m-funkcí a dynamických knihoven (DLL), které usnadňují komunikaci mezi Matlabem a rozličnými typy hardwarových zařízení
<i>Image Acquisition Toolbox</i>	rozšiřující modul Matlabu specializovaný na získávání obrazových dat a kontrolu hardwarových prostředků z prostředí Matlab
<i>Image Processing Toolbox</i>	soubor specializovaných funkcí pro veškerou práci s digitálním obrazem; zastřešuje prostorové transformace, morfologické operace, lineární filtry a mnoho dalších
<i>Signal Processing Toolbox</i>	sada nástrojů pro zpracování a analýzu signálu v digitální podobě; <i>SPT</i> je užít při zpracování zvukového signálu z převodníku
<i>Matlab Compiler</i>	překladač <i>m-kódu</i> Matlabu do jazyka C a C++ užívaný k produkci samostatně spustitelných aplikací, nebo dynamických knihoven, které lze referencovat ve vlastních aplikacích; dostupný je též kompilátor pro .NET kompatibilní jazyky (C#, VB.NET)
<i>Matlab Component Runtime</i>	Jelikož aplikace vyvíjené v Matlabu využívají celou řadu jeho vnitřních funkcí, jejich kompilace do každé jedné samostatné aplikace by bylo neefektivní. Proto představuje <i>MCR</i> samostatnou distribuci dynamicky volaných funkcí a je třeba jeho přednostní instalace před spuštěním aplikace Matlabu v případě, že Matlab není na hostitelském PC instalován. <i>MCR</i> je běhovým prostředím pro samostatnou aplikaci.

<i>RealTime Workshop</i>	rozšíření Simulink pro zjednodušenou kompilaci Simulink modelu, který je možné nasadit na specializovaném hardware pro zpracování v reálném čase
<i>xPC Target</i>	rozšíření Simulink; jde o řešení vhodné pro měření v reálném čase na tzv. cílovém PC, které slouží jako zařízení k získání dat; data samotná jsou zpracována na jiném PC; obě zařízení komunikují rozličnými způsoby včetně komunikace po síti LAN
<i>GUIDE</i>	Graphical User Interface Development Environment; nástroj Matlabu pro vývoj grafického uživatelského rozhraní nad m-kódem
<i>m-code</i>	z angl. matlab code; jde o zdrojový kód v jazyce Matlab
<i>callback</i>	obslužná funkce události; pomocí callback funkcí je realizována odezva uživatelského rozhraní na akce uživatele

Obsah

1	Úvod	7
2	Teoretická část	8
2.1	Elektrostatické zvlákňování	8
2.1.1	Parametry elektrostatického zvlákňování	9
2.2	Hladinové zvlákňování	10
2.3	Zpracování digitálního obrazu	17
2.3.1	Digitální reprezentace obrazu	17
2.3.2	Obecné postupy zpracování digitálního obrazu	22
2.3.3	Předzpracování obrazu	25
2.3.4	Analýza dvourozměrného obrazu v prostorové oblasti	29
2.3.5	Fourierova analýza	32
3	Experimentální část	33
3.1	Chemická část	33
3.2	Fyzikální část	33
3.2.1	Zvlákňovací sestava	33
3.2.2	Osvětlení scény	34
3.2.3	Prostředky získání dat	35
3.3	Programová část	40
3.3.1	Volba vývojového prostředí	40
3.3.2	Vytvořená aplikace	40
3.4	Výsledky měření a jejich interpretace	56
4	Závěr	62
	Užitá literatura:	65

1 Úvod

Tato práce se zabývá elektrostatickým zvlákňováním z volné hladiny polymeru. Jejím cílem je popis dosavadních postupů při zvyšování produktivity zvlákňovacích zařízení. Obsahem je také úvod do představ, které vedly k vytvoření robustní teorie samoorganizace volné hladiny polymeru ve stacionárním elektrickém poli. Pro ověření teoretických předpokladů byl zvolen experiment s tzv. lištovým spinnerem, který byl sledován pomocí digitální videokamery a zařízení pro sledování napětí elektrod. Z pořízených záznamů byly důležité poznatky získány pomocí analýzy obrazu. Teorie zpracování digitálního obrazu je v práci též zpracována. Získaná data jsou statisticky zpracována a podávají dobrý náhled na chování polymeru během zvlákňování. Protože lištový spinner není zařízení určené pro průmyslovou výrobu vláken a v tomto případě slouží pouze pro účely experimentu, je v závěru práce uvedena rozvaha o možné aplikaci zvolených metod během zvlákňování na zařízení NanospiderTM.

Nanovlákná jsou vlákna vyráběná z taveniny, nebo roztoku polymeru. Toto pojmenování nesou z toho důvodu, že jejich průměr nepřesahuje stovky nm. S jejich pomocí je možné získat materiály vynikajících vlastností, vhodných pro použití např. ve filtrační technice, tkáňovém inženýrství, řízeném dávkování léčiv, konstrukci tepelných a zvukových bariér. Využití v mnoha jiných oblastech je otázkou dalšího vývoje. Jednou z cest získání nanovláken je zvlákňování polymeru v elektrostatickém poli, kdy na polymer, umístěný mezi opačně nabitými elektrodami, působí silné elektrické pole. Toto pole způsobuje morfologické změny hladiny polymeru a při dosažení kritického napětí můžeme pozorovat tvarovou změnu kapalinových těles a vznik jemných proudů kapaliny směřujících k protilehlé elektrodě. Změny sférického tvaru kapky popsal jako první britský fyzik John Rayleigh. Další pozorování prováděl na počátku 20. století americký fyzik John Zeleny v práci [11]. Pokračovatelem v letech šedesátých se stal Brit G. I. Taylor, který popsal následující skutečnosti. Podle očekávání kapička posazená na vodorovném povrchu zaujímá tvar polokoule. Důvodem jsou vnitřní silové působení mezi částicemi kapaliny (tzv. povrchové napětí), podkladem a okolním prostředím. V elektrickém poli se tvar kapky mění v kuželový. Morfologické změny při elektrostatickém zvlákňování jsou výsledkem „konkurence“ povrchového napětí a vnějších elektrických sil (které mění elektrický náboj na povrchu kapaliny a působí proti soudržným silám). Pro tvar kapky ve fázi tvorby trysek je používán název „Taylorův kužel“ a právě z jeho vrcholu je hmota kapky uvolňována. Pokud na cestě od povrchu k protilehlé elektrodě,

kolektoru, nedojde k rozprášení kapaliny, ale k vzniku lineárních vláken, pak tento jev označujeme jako elektrostatické zvlákňování, zkráceně elektrospinning.

O historicky prvním zařízení na výrobu vláken metodou zvlákňování v elektrostatickém poli (dále jen electrospinning) svědčí patent [12] z roku 1934, který registroval Anton Formhals. V patentu je popisováno zařízení na výrobu jemných vláken z roztoku celulózy s kyselinou octovou, založené na soustavě uspořádaných jehel a kolektoru (proti elektrodě). Následně po zveřejnění zmíněného patentu se o podobný princip pokoušeli mnozí další konstruktéři. Nicméně elektrospinning se jako průmyslový způsob výroby vláken neujal. Důvodem byla vždy nízká produktivita výroby.

V posledních letech zažívá elektrospinning jistou formu renesance v podobě úspěchů na poli zvyšování produktivity strojů pro elektrostatické zvlákňování (dále jen spinnerů) a snahy o výrobu nanovláken v průmyslovém měřítku.

2 Teoretická část

Teoretická část práce se zabývá dvěma hlavními oblastmi. V první části je nastíněn proces elektrostatického zvlákňování z hlediska chemického a fyzikálního. Druhá část je věnována základním principům zpracování a analýzy obrazu v jeho digitální reprezentaci.

2.1 Elektrostatické zvlákňování

Pro klasické zvlákňování v elektrostatickém poli je charakteristické zpracováním jedné kapky umístěné na jedné kapiláře - jehle. Jde vždy o laboratorní zařízení s velmi nízkou produkcí. Snahy o zvýšení produktivity vedly například k prostému násobení počtu trysek paralelně vedle sebe a tvorbě jehlových polí. Tento přístup skutečně vede k vyšší produkci úměrné počtu trysek, ovšem nedostatky převládají. Jde hlavně o zanášení trysek polymerem, vyšší složitost zařízení, nutnost přesného dávkování jednotlivých kapilár a další. Velký průlom v produkci, a vlastně první průmyslově použitelnou metodu přinesla technologie NanospiderTM, vyvinutá díky spolupráci týmu vědců Katedry netkaných textilií Technické Univerzity v Liberci, pod vedením prof. Jirsáka, s libereckou firmou Elmarco. Tato patentovaná [7] technologie opouští klasický přístup zvlákňování z kapky na konci jehly. Principem je vznik trysek z volné hladiny polymeru. Volná hladina je získána na povrchu válce brodícího se polymerní lázni. Tento přístup podstatně redukuje počet pohyblivých mechanických částí stroje. Poskytuje vysokou produkci kvalitních, stejnoměrných vláken, a zároveň neklade

nadměrné požadavky na údržbu. Cesta, kterou tato technologie razí, je jistě zajímavá. Díky tomu, že je založena čistě na spontánní organizaci hmoty polymeru do podoby vláken, vyvstala potřeba hlubšího poznání fyzikální podstaty procesu formování trysky, její dynamiky, přenosu hmoty z roztoku na kolektor a také vlivu okolních podmínek.

2.1.1 Parametry elektrostatického zvlákňování

Při elektrostatickém zvlákňování polymeru hraje podstatnou roli řada fyzikálních a chemických vlivů. O několika hlavních parametrech stručně pojednávají následující odstavce.

Napětí elektrod

Napětí elektrod a tím vyvolané elektrické pole v okolí spinneru je při zvlákňování rozhodujícím kritériem. Vysoké napětí způsobuje hromadění náboje shodného znaménka na povrchu polymeru. Vznikající odpudivé síly jsou příčinou morfologických změn na hladině. Kapka na ústí kapiláry změní tvar v tzv. Taylorův kužel; popis chování hladiny polymeru je obsažen v kapitole 2.2. Přitažlivé síly zapříčiňují transport hmoty polymeru směrem ke kolektoru a ukládání vláken nahodile do vrstev. Profil elektrostatického pole ovlivňuje trajektorii vznikajícího vlákna a je proto důležitým předmětem studia.

Kromě intenzity a charakteru elektrického pole má, podle [3], podstatný vliv také kvalita samotného polymeru. Mezi klíčové vlastnosti patří viskozita, povrchové napětí a vodivost roztoku.

Povrchové napětí

Podle kapitoly 2.2 je iniciace zvlákňování, mimo jiné, závislá na překonání kapilárních sil vnějším elektrostatickým polem. Zároveň vysoká hodnota povrchového napětí, které nutí kapalinu zaujmout tvar s nejmenším možným povrchem, způsobuje rozpad vláknotvorné trysky na cestě polymeru od hladiny ke kolektoru v kapky. Žádoucí je z obou těchto důvodů snižování povrchového napětí roztoku polymeru. Toho může být docíleno přidávkem povrchově aktivní látky.

Viskozita

Jedním z faktorů, ovlivňujících viskozitu polymeru, je jeho molekulová hmotnost. Ta je měrou délky polymerního řetězce. Obecně platí, že roztok vysokomolekulárního polymeru má vyšší viskozitu než roztok polymeru s nižší molekulovou hmotností a stejné koncentraci.

Během zvlákňování je právě délka řetězců a míra jejich zapletení příčinou vzniku vláknenných útvarů a nikoli kapek. Podstatný je neustálý souboj jednotlivých parametrů na molekulární úrovni, který je příčinou makroskopického chování roztoku. Dlouhé zapletené řetězce polymeru mají snahu zaujmout v rozpouštědle větší objem a nelze je snadno oddělit. Naproti tomu povrchové napětí přispívá k snižování objemu kapaliny. Z tohoto důvodu v monomerních kapalinách převládá vliv povrchového napětí, které je příčinou vzniku kapének namísto vláknenných útvarů. Další vliv na viskozitu roztoku má koncentrace polymeru. Zvýšená koncentrace, stejně jako délka řetězců, vede k vyšší míře vzájemného propletení řetězců polymeru. To způsobuje snížení jejich pohyblivosti a zvýšení viskozity roztoku. Vláknena z roztoku vyšší koncentrace mívají též větší průřez. Dostatečná viskozita je nutným předpokladem pro správný průběh zvlákňování. Její vysoká hodnota ovšem přináší jisté problémy. Z příliš koncentrovaného roztoku se může rozpouštědlo odpařit příliš rychle, ještě na povrchu spinneru, a proces zvlákňování neproběhne.

Vodivost

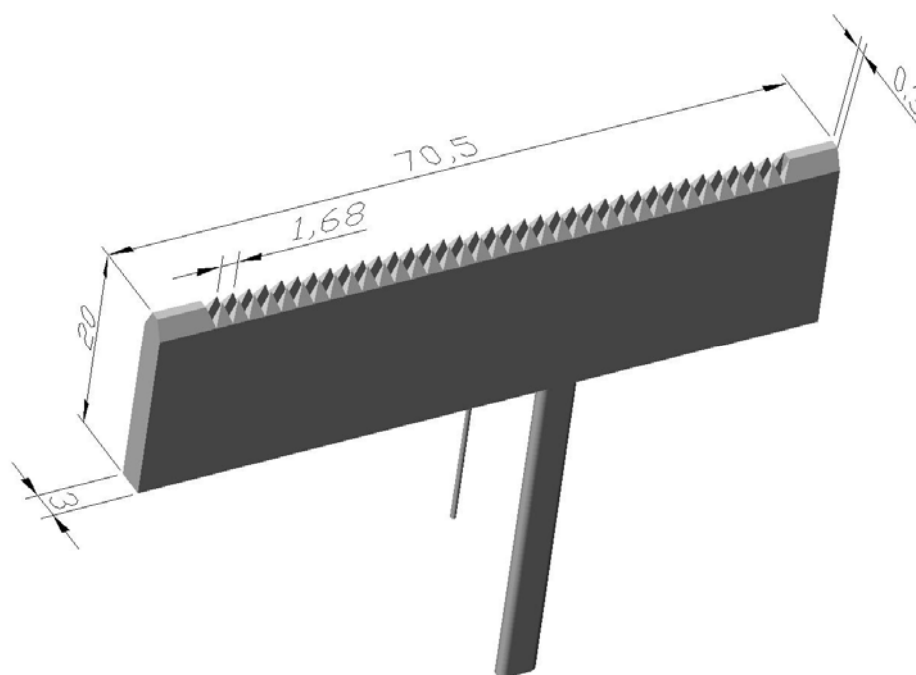
Elektrostatické zvlákňování je důsledkem odpuzivých sil na povrchu polymeru a zároveň přitažlivých sil vůči kolektoru. Zvýšení vodivosti roztoku přidáním volných iontů dovoluje zvýšenou koncentraci nábojů na jeho povrchu polymeru a vede k zlepšení vlastností z hlediska zvlákňování. Zvýšení počtu volných iontů je nejjednodušeji dosaženo přidáním soli do vodného roztoku. Díky vyšší koncentraci volných iontů, zvýšené vodivosti roztoku, je možné snížení potřebného napětí na elektrodách. Zároveň jsou produkována vlákna tvarově stabilnější, menších průměrů a pravděpodobnost vzniku kapky je podstatně nižší.

2.2 Hladinové zvlákňování

Během elektrospinningu na stroji NanospiderTM jsou vlákna vytahována z povrchu válce pokrytého roztokem polymeru. Tento proces nazýváme bezjehlové (needle-less) zvlákňování, nebo také *zvlákňování hladinové*.

V porovnání s klasickými jehlovými spinery, které byly konstruovány a dobře popsány v 2. polovině 20. století Taylorem v práci [8], pro popis hladinového zvlákňování dosud širší teorie chyběla. Podstatnou změnou představuje článek [1], který popisuje vznik fyzikálních dějů při elektrostatickém zvlákňování z volné hladiny polymeru. Teoretické předpoklady a z nich vycházející rovnice se opírají o experimenty provedené na tzv. lištovém spinneru. Lištový spinner poskytuje značné zjednodušení problému, neboť hladina polymeru v úzkém kanálu

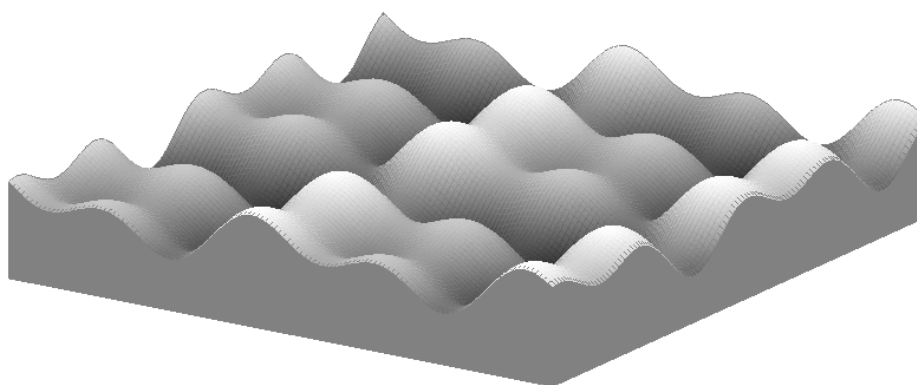
představuje lineární útvar položený do vodorovné souřadnicové osy. Model lištového spinneru je znázorněn na Obr. 1.



Obr. 1: Model lištového spinneru použitého pro ověření teorie elektrostatického zvlákňování z volné hladiny polymeru.

Vlnění hladiny je reprezentováno pohybem ve směru vertikálním. Vyslovená teorie nepostihuje pohyb hladiny do třetího rozměru, ten je však použitím lištového spinneru téměř dokonale potlačen.

Přirozeným projevem pohybu mechanického systému, za který můžeme považovat volnou hladinu kapaliny, je nepatrné zvlnění.



Obr. 2: Počítačový model přirozeného zvlnění hladiny.

Vezměme nejjednodušší případ, kdy se vlna na hladině může rozpohybovat jen v jednom směru – to je případ kapaliny v úzké štěrbině. A nyní se zaměříme na jeden bod hladiny právě

v okamžiku jeho rovnovážné polohy. Jakákoli výchylka z tohoto bodu je příčinou vzniku síly, jejíž účinek směřuje proti nastalé změně. Zde je jistá podobnost s modelem hmotného bodu zavěšeného na pružině. Při vychýlení z rovnovážné polohy působí pružina silou

$$F = -ky \quad (1)$$

kde k je tuhost a y výchylka. Z 2. Newtonova pohybového zákona známe, že

$$F = m \frac{d^2 y_{(t)}}{dt^2} \quad (2)$$

a pohybová rovnice je ve tvaru

$$m \frac{d^2 y_{(t)}}{dt^2} + ky_{(t)} = 0 \quad (3)$$

Řešením lineární diferenciální rovnice s konstantními koeficienty

$$\frac{d^2 y_{(t)}}{dt^2} + \omega^2 y_{(t)} = 0, \quad (4)$$

kde $\omega^2 = \frac{k}{m}$, získáme

$$y_{(t)} = C_1 \cos(\omega t) + C_2 \sin(\omega t) \quad (5)$$

Toto řešení je možné zapsat jako

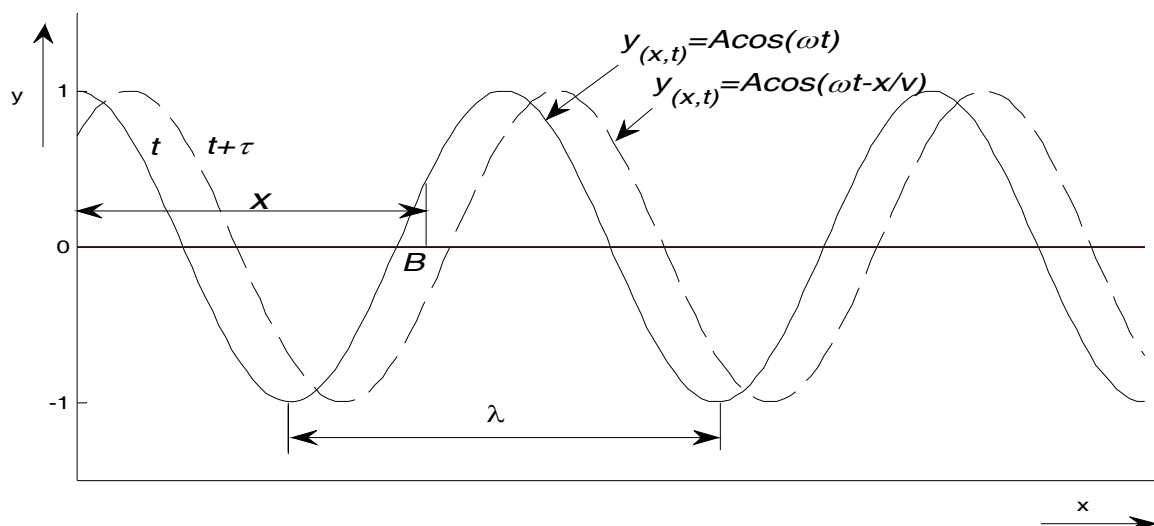
$$y_{(t)} = A \cos(\omega t + \varphi), \quad (6)$$

kde $\tan(\varphi) = -\frac{C_2}{C_1}$, $A = \sqrt{C_1^2 + C_2^2}$. Podle [2] můžeme polohu bodu na hladině y zapsat jako

reálnou složku komplexní proměnné

$$y_{(t)} = A e^{i(\omega t + \varphi)} \quad (7)$$

Uvedené rovnice popisují skutečnost, že v okolí rovnovážného stavu koná bod harmonické kmity s frekvencí ω a amplitudou A . Hladina je tvořena nekonečným množstvím těchto oscilátorů, přičemž každý z nich rozkmitává své nejbližší okolí a oscilace se tak šíří hmotným prostředím – vzniká příčné vlnění. V popisovaném zjednodušeném případě se vlnění šíří jen směrem, který představuje vodorovná osa x souřadného systému.



Obr. 3: Postupné vlnění jednorozměrné vlny ve dvou časových okamžicích.

Vlnění se šíří rychlostí v , takže do bodu B , vzdáleného od počátku o x , dospěje za dobu $\tau = \frac{x}{v}$. Výchylka každého místa postupné vlny je proto funkcí polohy a času:

$$y_{(x,t)} = A \cos \omega(t - \tau) = A \cos \omega \left(t - \frac{x}{v} \right),$$

$$\omega = \frac{2\pi}{T} \quad (8)$$

Volbou záporného ω získáme ekvivalentní vztah: $y_{(x,t)} = A \cos \left(\omega \frac{x}{v} - \omega t \right)$. Záporná úhlová frekvence přijde vhod dále. Protože $\omega \frac{x}{v} = \frac{2\pi}{T} \frac{x}{v} = \frac{2\pi}{\lambda} x = kx$, může předchozí vztah zapsat takto:

$$y_{(x,t)} = A \cos(kx - \omega t), \quad (9)$$

$$y_{(x,t)} = A e^{i(kx - \omega t)}, \quad (10)$$

kde k je *vlnový vektor*. Vlnový vektor má směr šířící se vlny a velikost $|\vec{k}| = \frac{2\pi}{\lambda}$. V tomto případě má jen složku x , směry do ostatních os jsou nulové $\vec{k} = \left(\frac{2\pi}{\lambda}, 0, 0 \right)$ a je tak správnější jej nazývat *vlnočeta*.

Teorie samoorganizace trysek na volné hladině polymeru, uvedená v [1], vychází kromě vztahu (10) také z *dispersního zákona* pro čtverec úhlové frekvence jednorozměrné vlny pod vlivem elektrického a gravitačního pole:

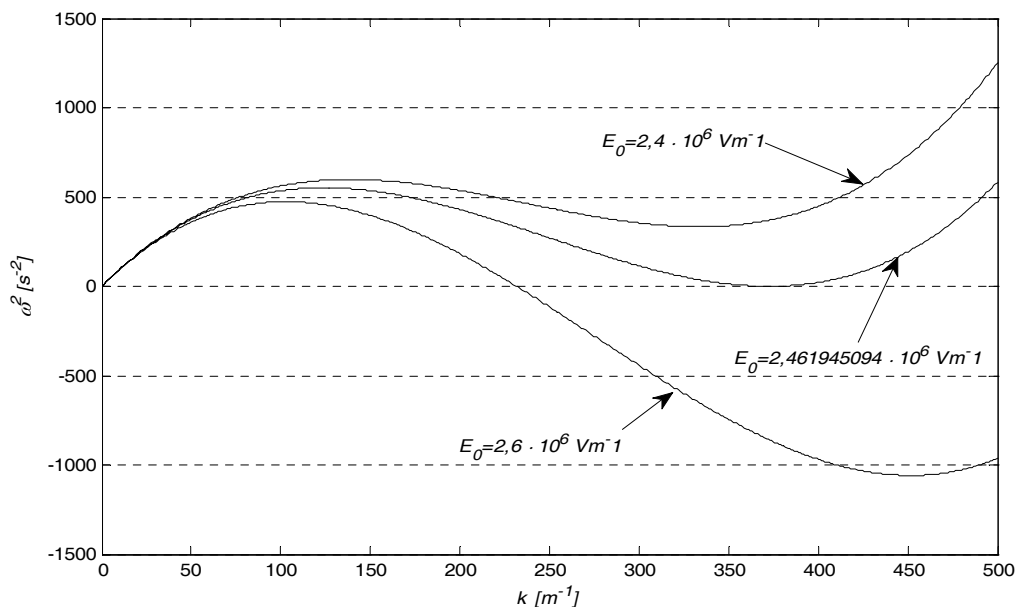
$$\omega^2 = \left(\rho g + \gamma k^2 - \varepsilon E_0^2 k \right) \frac{k}{\rho}, \quad (11)$$

kde je ρ – hustota, g – gravitační zrychlení, γ – povrchové napětí, k – vlnový vektor, ε – permitivita, E – intenzita elektrického pole.

Z rovnice (11) je zřejmé, že hodnota úhlové frekvence závisí na hodnotách gravitačního pole, kapilárních sil a pole elektrického. V případě, že se hodnota intenzity elektrického pole E rovná hodnotě kritické E_0 , pak je úhlová frekvence nulová. Při dalším zvyšování intenzity přejde kvadrát ω v hodnoty záporné. Taková úloha je řešitelná pouze v oboru komplexních čísel a to tak, že ω je ryze imaginární. Dosazením $i\omega$ do (10) a vyjádřením $i^2 = -1$ můžeme psát:

$$y_{(x,t)} = Ae^{i\omega t} e^{i(kx)} \quad (12)$$

Úhlová frekvence tak změnila svůj fyzikální význam. Ztráta časového parametru v druhém exponentu znamená přechod od postupného vlnění k vlnění stojatému s bezmezne rostoucí, časově závislou amplitudou $Ae^{i\omega t}$.



Obr. 4: Závislost kvadrátu úhlové frekvence ω na vlnočtu k pro tři různé hodnoty intenzity elektrického pole E_0 ; $\gamma = 72 \text{ mN.m}^{-1}$, $\rho = 1000 \text{ kg.m}^{-3}$, $\varepsilon = 8.854 \cdot 10^{-12} \text{ s}^4 \text{ A}^2 \text{ kg}^{-1} \text{ m}^{-3}$.

Kritickou hodnotu vlnového vektoru můžeme získat z disperzního zákona tak, že položíme $\omega^2 = 0$.

To vede na řešení kubické rovnice $(\rho g + \gamma k^2 - \varepsilon E_0^2 k) \frac{k}{\rho} = 0$. Jedním kořenem je $k_1 = 0$.

Vynásobením rovnice $\frac{\rho}{k}$ získáváme rovnici kvadratickou,

$$\gamma k^2 - \varepsilon E_0^2 k + \rho g = 0 \quad (13)$$

Obr. 4 znázorňuje $\omega = \omega_{(k, E_0)}$ pro tři různé hodnoty E_0 . Diskriminant rovnice (13) je

$$D = \varepsilon^2 E_0^4 - 4\gamma\rho g \quad (14)$$

Při zjišťování kritické hodnoty vlnového vektoru hledáme podle Obr. 4 dvojnásobný kořen rovnice (13). Příznivá situace nastane v rovnici (14) pro $D = \varepsilon^2 E_0^4 - 4\gamma\rho g = 0$. Pak řešení rovnice (13) odpovídá:

$$k_{2,3} = \frac{\varepsilon E_0^2}{2\gamma} \quad (15)$$

a hodnota příslušné intenzity elektrického pole je:

$$E_0 = \sqrt[4]{\frac{4\gamma\rho g}{\varepsilon^2}} \quad (16)$$

Kapilární délka, $a = \sqrt{\frac{\gamma}{\rho g}}$, je často užívaným parametrem v koloidní chemii. Pomocí

kapilární délky můžeme rovnici (16) přepsat na tvar $\frac{1}{2}\varepsilon E_0^2 = \frac{\gamma}{a}$. Práce [1] zavádí

bezrozměrné tzv. elektrospiningové číslo:

$$\Gamma = \frac{a\varepsilon E_0^2}{2\gamma}, \quad (17)$$

které je interpretováno tak, že $\Gamma < 1$ je podmínkou stability povrchových vln.

K zjištění kritické, tedy minimální vzdálenosti mezi jednotlivými tryskami během elektrospinningu při dané intenzitě elektrického pole, vede nalezení minima kvadrátu kruhové

frekvence ω , viz rovnice (11). První derivace rovnice (11) podle k vede na rovnici

$$\frac{\partial \omega^2}{\partial k} = \frac{3\gamma}{\rho} k^2 - \frac{2\varepsilon E_0^2}{\rho} k + g = 0.$$

Jejím řešením je:

$$k_{(1,2)} = \frac{\varepsilon E_0^2 \pm \sqrt{(\varepsilon E_0^2)^2 - 3\gamma\rho g}}{3\gamma} \quad (18)$$

Smysl má vyšší hodnota k , protože hledáme minimum funkce, viz Obr. 4. Zároveň musí být v tomto bodě splněna podmínka $\omega^2 < 0$. Vztah mezi vlnovou délkou λ a vlnovým vektorem k , zní: $\lambda = \frac{2\pi}{k}$. Kritickou vlnovou délkou určuje vztah (19):

$$\lambda_c = \frac{6\pi\gamma}{\varepsilon E_0^2 + \sqrt{(\varepsilon E_0^2)^2 - 3\gamma\rho g}}, \quad (19)$$

V článku [1] a v této práci je využito vzájemného vztahu bezrozměrných veličin $\frac{\lambda}{a}$, Γ , který je platný pro všechny kapaliny takto:

$$\frac{\lambda}{a} = \frac{3\pi}{\Gamma + \sqrt{\Gamma^2 - \frac{3}{4}}}. \quad (20)$$

Výsledky dosud provedených experimentů byly přes četná zjednodušení v dobrém souladu s nalezenými vztahy. Teorie uvedená v článku [1] tak představuje potenciál nalezení optimálních parametrů pro konstrukci bez-jehlových zvlákňovacích zařízení.

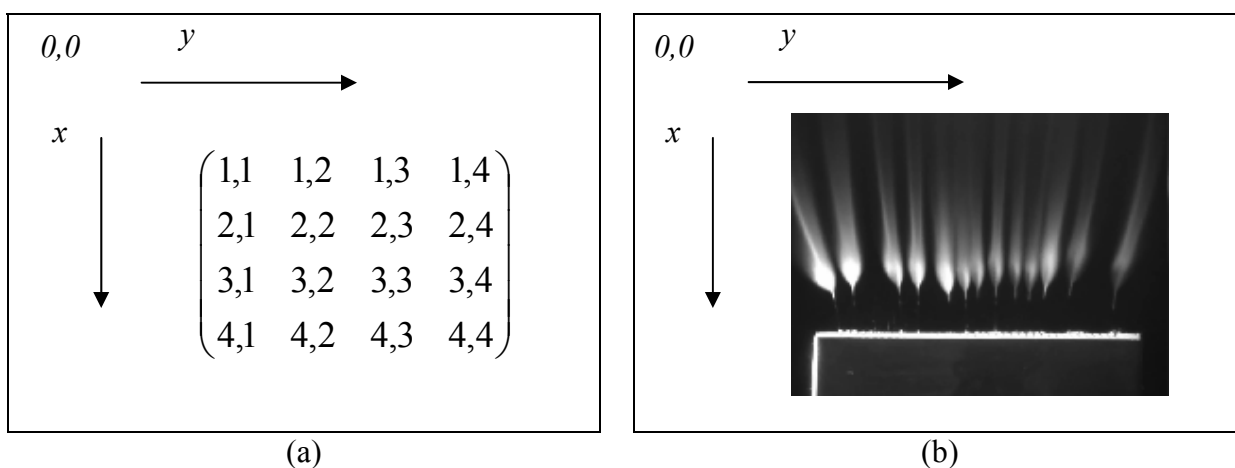
Získání experimentálních dat je v případě elektrospiningu komplikovanou úlohou. Mezi elektrody nelze vložit žádné mechanické měřidlo, nebo sondu, která by vypovídala o charakteru trysek a lokální hodnotě intenzity elektrického pole. Jedinou dosud rozumně aplikovatelnou metodou bylo zachytávání procesu zvlákňování kamerou a pozdější ruční zpracování. Při zachytávání *miniDV* kamerou je pořízení 25 snímků za vteřinu. V lidských silách není možné takový objem dat za rozumnou dobu zpracovat. Nehledě na skutečnost, že sledování statických snímků neodhalí dynamiku procesu - např. chování trysek po náhlém zvýšení napětí elektrod.

2.3 Zpracování digitálního obrazu

Zpracování digitálního obrazu je rozsáhlá disciplína, jejíž vznik byl podmíněn rozvojem výpočetní techniky. Využívá velmi sofistikované metody pro manipulaci a získání logických informací z obrazových dat. V následující teoretické části jsou probrány základy postupu získání a uchování digitálního obrazu. Některé užité metody obrazové analýzy jsou vysvětleny a doloženy na příkladech. V textu je jako oddělovací znak desetinného místa v souladu s notací Matlabu použita 'tečka'.

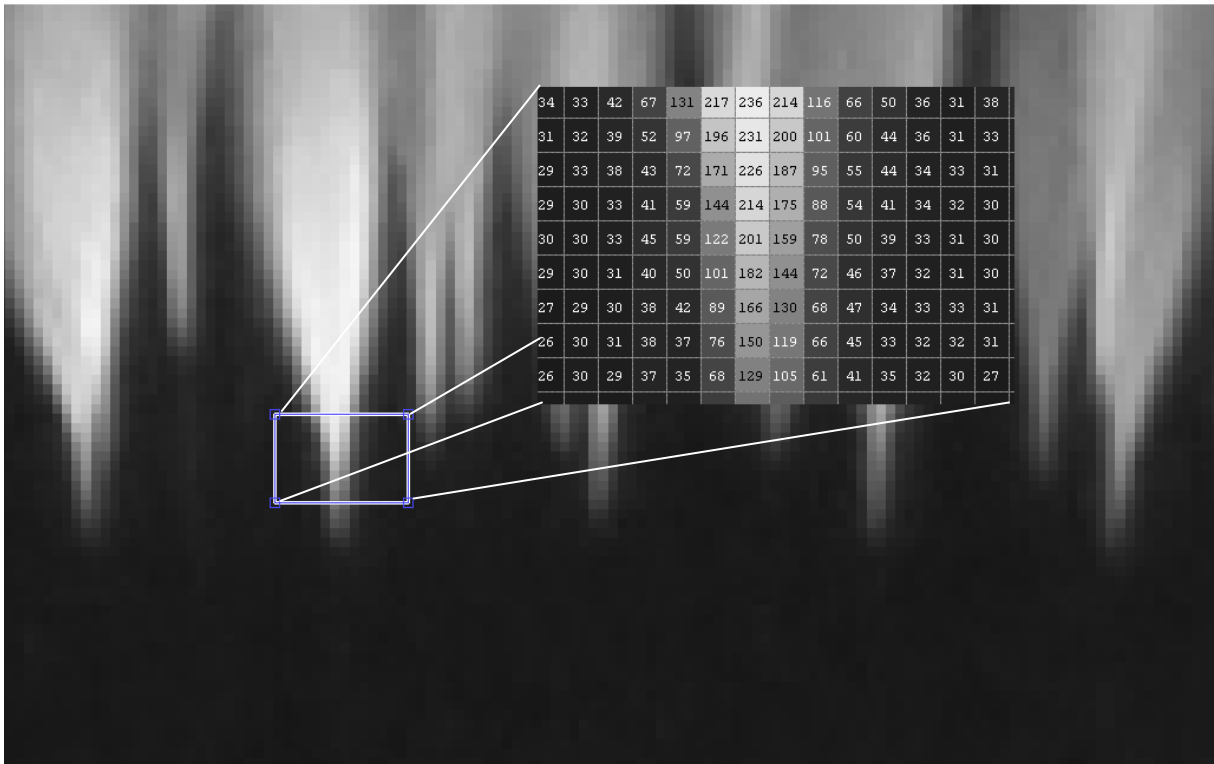
2.3.1 Digitální reprezentace obrazu

Obraz definujeme jako funkci dvou proměnných, $f(x,y)$, kde x, y představují jednotlivé prostorové souřadnice místa v obraze. Protože prvky matice označujeme zpravidla indexem řádku následovaný indexem sloupce a první prvek matice se nachází v levém horním rohu matice, reflektují souřadnice digitálního obrazu toto značení. Indexace prvků matice a na skutečném obraze je vidět na Obr. 5.



Obr. 5: (a) Zobrazení polohy a orientace souřadných os, (b) příklad indexace polohy v digitálním obraze.

Funkční hodnotou $f(x,y)$ je intenzita světla v daném místě, jak je dobře vidět na Obr. 6. Pro konkrétní "bod" v digitálním obraze se nejčastěji používá termínu *pixel* (z angl. *picture element*). V tomto smyslu můžeme dále obraz rozdělit na *barevný* a *intenzitní* (šedotónový). Barevný obraz je definován stejně jako obraz intenzitní – prostorovou funkcí. Rozdíl spočívá v použití funkce intenzity pro každou barevnou složku zvlášť a jejich vzájemným překrytím. Amplitudou není stupeň šedé, ale např. u RGB obrazu stupeň červené, zelené a modré barvy.



Obr. 6: Digitální reprezentace šedotónového obrazu.

Prostorové souřadnice zachytávané scény jsou spojité veličiny. Stejně tak hodnota jasu (funkční hodnota, amplituda). Použitím číslicového zaznamenávacího zařízení nastává proces diskretizace hodnot. *Vzorkováním* provádíme digitalizaci souřadnic, *kvantizací* provádíme digitalizaci jasové složky. Díky těmto dvěma procesům získáme hodnoty všech tří veličin $x, y, f(x,y)$ konečné a diskrétní. Výsledná data nazýváme digitální reprezentací obrazu. Tato data představují matici reálných hodnot. Velikost matice je dána vertikálním a horizontálním rozlišením obrazu. Např. obrazovku PC o rozlišení 1024x768 v 24bit hloubce barev je možné reprezentovat třemi maticemi o rozměrech 768 řádků x 1024 sloupců, přičemž jas každé barevné složky je kvantizován do rozsahu hodnot 0-255.

Vzorkování a kvantizace

Následující ilustrace prezentují obrazy vzniklé změnou parametrů *vzorkování* (rozlišení) a *kvantizace* (barevné hloubky) během zachycení scény.



(a)

(b)

(c)

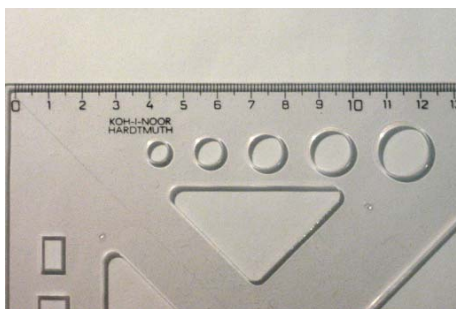
Obr. 7: Kvantizace během digitalizace obrazu; (a) 256 jasových úrovní, (b) 16 jasových úrovní, (c) 4 jasové úrovně.

Zajímavou skutečností je, že lidské oko příliš nevnímá rozdíl mezi obrazy *kvantizovanými* nad hranicí šestnácti úrovněmi jasu. Pokud bychom tedy považovali 16 úrovní jasu za dostatečnou jasovou hloubku, potřebovali bychom k uchování jednoho obrazového bodu pouze 4 *bity* ($16 = 2^4$) paměti. Nejnižším číselným datovým typem Matlabu je osmibitový *integer*. Proto je při zpracování obrazu vždy používáno 256 jasových úrovní.

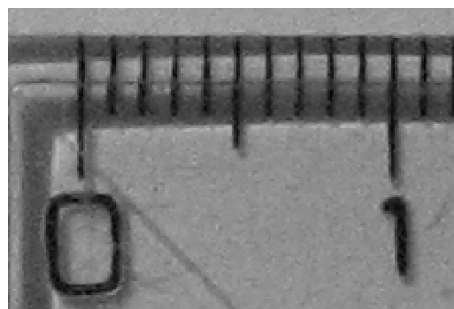
Vzorkováním podle obou souřadných os měníme rozlišení obrazu. Vyšší vzorkovací frekvence dovoluje zachycení větších detailů. Při výběru vzorkování pomáhá tzv. *Shannonova věta*, která říká, že minimální vzorkovací frekvence musí být alespoň dvakrát vyšší, než je nejvyšší zajímavá frekvence ve spojitě scéně.

Pokud potřebujeme měřit vzdálenost v obraze, je vzorkování zásadním parametrem. Fotografie na Obr. 8 ilustrují zachycení jedné scény při různých vzorkovacích frekvencích. Měříme milimetrové značky na vzdálenosti jednoho centimetru. Značky se tedy objevují s frekvencí 10/cm. Minimální potřebná horizontální frekvence je 20 px/cm.

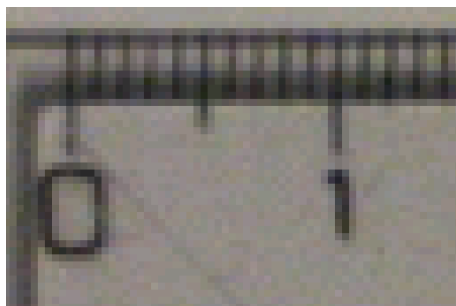
Evidentně při nízké vzorkovací frekvenci není možné detaily rozlišit a měření vzdálenosti není možné. V této práci je snímána scéna o šíři 7cm *miniDV* kamerou o horizontálním rozlišení 720 px. Tomuto odpovídá vzorkovací frekvence zhruba 103 px/cm a teoreticky je možné rozlišit detaily s frekvencí 50/cm. Perioda je reciprokou hodnotou frekvence. Rozlišitelná perioda, a tedy vzdálenost, je $1/50=0,02$ cm. V praxi je díky zašumění a neostrosti zachyceného obrazu nejnižší rozpoznatelná perioda větší, přesto je možné považovat horizontální rozlišení 720 obrazových bodů za vyhovující pro danou aplikaci. Během pořizování záznamu je ale nutné brát tyto skutečnosti v úvahu a snažit se pomocí přiblížení kamery zachytit scénu v největším možném přiblížení.



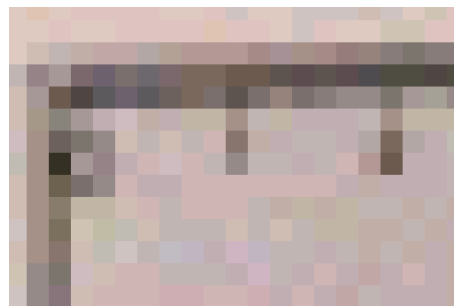
(a)



(b)



(c)



(d)

Obr. 8: *Vzorkování při digitalizaci obrazu; (a) zachycovaná scéna, (b) vzorkování 155 px/cm, (c) vzorkování 38 px/cm, (d) vzorkování 15 px/cm.*

Definice vzdálenosti a okolí

Při digitalizaci je spojitý obraz vzorkován do rastru obrazových bodů. Nejčastěji je rastr pravoúhlý a jednotlivé obrazové body jsou ekvidistantně vzdálené. Okolí pixelu na dvourozměrné mříži může být definováno čtyřmi (*čtyřokolí*) nebo osmi (*osmiokolí*) nejbližšími pixely, podle toho zda jsou jako okolní pixely chápány i prvky na diagonále.

Existuje více způsobů, jak definovat vzdálenost jednotlivých prvků v pravoúhlé mříži.

Mějme dva body $f(x_1, y_1), f(x_2, y_2)$:

Euklidovská vzdálenost:

Euklidovská vzdálenost je přímá spojnice dvou bodů. Z geometrie je pro rovinu definována jako:

$$L_E = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (21)$$

2.83	2.24	2.00	2.24	2.83
2.24	1.41	1.00	1.41	2.24
2.00	1.00	0	1.00	2.00
2.24	1.41	1.00	1.41	2.24
2.83	2.24	2.00	2.24	2.83

Obr. 9: Euklidovská vzdálenost

Nevýhodou je, že výsledkem není celočíselná hodnota, a to neodpovídá charakteru diskrétně rozdělených bodů v rastru.

Vzdálenost v městských blocích

Vzdálenost v městských blocích znamená nejmenší počet kroků, který je třeba učinit na pravoúhlé cestě mezi dvěma body. Nejmenší jednotkou vzdálenosti je jeden pixel. Matematicky je tato vzdálenost definována jako:

$$L_4 = |x_2 - x_1| + |y_2 - y_1| \quad (22)$$

4.00	3.0	2.00	3.00	4.00
3.00	2.00	1.00	2.00	3.00
2.00	1.00	0	1.00	2.00
3.00	2.00	1.00	2.00	3.00
4.00	3.00	2.00	3.00	4.00

Obr. 10: Vzdálenost v městských blocích

Za okolní obrazové body jsou považovány čtyři nejbližší pixely. Vzdálenost čtyř bodů na diagonálách je 2.

Vzdálenost na šachovnici

Pokud budeme chápat okolí pixelu jako nejbližších osm sousedních bodů, pak je jejich vzdálenost jedna. To odpovídá možnostem pohybu krále na šachovnici. Definice je tato:

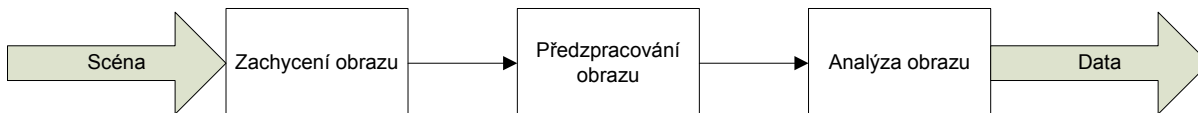
$$L_8 = \max\{|x_2 - x_1| + |y_2 - y_1|\} \quad (23)$$

2.00	2.00	2.00	2.00	2.00
2.00	1.00	1.00	1.00	2.00
2.00	1.00	0	1.00	2.00
2.00	1.00	1.00	1.00	2.00
2.00	2.00	2.00	2.00	2.00

Obr. 11: Vzdálenost na šachovnici

2.3.2 Obecné postupy zpracování digitálního obrazu

Obrazová data lze v počítači upravovat a transformovat použitím matematických, množinových a logických operací s cílem získat z obrazu požadovanou informaci. Těmito operacím se říká *předzpracování obrazu*, *analýza obrazu* a *strojové vidění*. Přestože mezi těmito pojmy není striktní hranice, nejde o tytéž. Rozdíl je na úrovni přístupu k datům.



Obr. 12: Schéma postupu při analýze informací z digitálního obrazu.

Předzpracováním obrazu rozumíme úpravy na nejnižší úrovni, tedy takové, které si nekladou za cíl zjistit charakter zachycené informace. Operace se zařazují před vlastní *analýzu obrazu*. Cílem je zvýraznit a zlepšit čitelnost informací. Tím pádem restaurovat ze zachyceného obrazu co nejvíce dat obsažených v původní scéně. Jde např. o filtry vedoucí k potlačení šumu, zvýšení kontrastu, zvýšení ostroty, vyrovnaní histogramu a další. Charakteristický pro tyto operace je fakt, že na vstupu i výstupu operace je obraz.

Analýza obrazu představuje střední úroveň zpracování a je tak mezistupněm mezi základními operacemi a strojovým viděním. Procesy jsou komplexnější povahy, zahrnují segmentaci obrazu (rozlišení objektů od pozadí, rozdělení jednotlivých objektů od sebe) a rozpoznávání

objektů (jejich klasifikace podle charakteru). Operace tedy různými způsoby klasifikují objekty v obraze podle jejich vlastností. Například pro nalezení objektu vlákněného charakteru lze pro každý objekt v obraze zjistit jeho délku, tloušťku a orientaci. Často je vhodné obraz rozdělit do menších částí, zpracovávaných samostatně. Charakteristické pro tuto úroveň zpracování je, že na vstupu figuruje obraz, zatímco na výstupu získáváme vlastnosti objektů. Jde typicky o hranice objektů, jejich těžiště, pozici v obraze a podobně.

Nejvyšší možnou interpretaci obrazové informace zveme *strojové vidění*, které se snaží o pochopení objektů a dějů ve scéně. Úlohu často sehrává paměť o dříve zpracovaných snímcích. Příkladem je zpracování sekvence obrazů v čase a predikce pohybu, tělesa na základě předchozího 'chování', nebo čtení skenovaného textu. Obecně jsou tyto úlohy velmi obtížně definovatelné. Pro člověka je vidění základním smyslem, na který se spoléháme podvědomě. Ovšem postavit systém s alespoň podobnými schopnostmi je úkol obtížný. Je třeba si uvědomit, že člověk se neustále učí a díky paměti je schopen přiřazovat objektům ve scéně podle vlastní zkušenosti nějaký smysl. Konstrukce učenlivých strojů je ale samostatnou vědou. Proto je obecně známo, že se v této oblasti nedaří dosahovat takových pokroků, jaké byly dříve očekávány.

Vždy platí, že úspěšnost každé "vyšší" operace je podmíněna úspěchem předchozího zpracování a nedostatky pořízeného (např. neostře) záznamu, není možné žádnou operací dokonale nahradit. V této práci zacházíme s obrazem do střední úrovně zpracování, tedy analýzy dvourozměrného obrazu.

Obvyklé problémy při zpracování digitálního obrazu a jejich řešení

3D -> 2D transformace

Pozorování prostorové scény a zachytávání jedním zařízením přináší ztrátu perspektivy. Díky tomu se stává velmi obtížnou úlohou stanovit hloubku obrazu, a tedy vzdálenost objektů v rovině kolmé k rovině snímání. Vždy platí, že ani ta nejlepší úprava neodstraní vady způsobené nešikovným záznamem. Řešení tohoto problému spočívá ve volbě vhodného snímání scény. Konkrétně pro nalezení vzdálenosti v horizontálním směru je třeba kameru postavit kolmo k rovině, ve které je třeba vzdálenost stanovit. Tak je možné vyhnout se pozdějším úpravám perspektivy.

Velké množství dat

Běžná digitální kamera zachycuje barevné snímky s frekvencí 25 Hz, každý v rozlišení 720x576. To představuje tři matice velikosti 576x720, přičemž každý element těchto matic je kvantizován s dvojitou přesností (double precision). Pro jeden snímek nekomprimovaného videa v rozlišení *miniDV* je třeba v paměti alokovat $720 \cdot 576 \cdot 3 \cdot 64 = 79\,626\,240$ bitů. To představuje téměř 1MB dat potřebných k uložení jednoho barevného snímku. Protože zpracování barevného obrazu je podobné zpracování obrazu intenzitního – jen pro každou barevnou složku zvlášť, a díky tomu že barva obrazu s sebou často nenese podstatnou informaci o dění ve scéně, přistupujeme k předběžnému odstranění barevné složky a sloučení intenzit jednotlivých barev do odstínů šedé. Pro další zpracování zároveň často postačuje 256 jasových úrovní (od černé = 0 do bílé = 255), tedy datový typ uint8. Potřebná paměť k uchování takového obrazu je $576 \cdot 720 \cdot 8 = 3\,317\,760$ bitů ~ 414 kB. Popsané úpravy pro zvolenou aplikaci ztrátu informace nepředstavují, a proto jsou užity v rámci předzpracování videa.

Přítomnost šumu

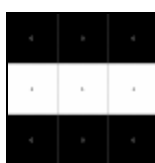
Šum znamená jakýkoli údaj ve vstupním signálu, který nijak nevypovídá o vlastnostech pozorovaných objektů. Jeho přítomnost však podstatné informace zastírá a znesnadňuje rozlišitelnost objektů. V oblasti zpracování obrazu existuje celá řada metod vedoucích k potlačení šumu, jako je prahová funkce nebo morfologické operace, které jsou zmíněny dále. Provedené operace ale mívají vliv na celé spektrum signálu, a tedy i na informace podstatné. Hledání správných metod a vyladění jejich parametrů je poměrně pracnou činností. Do úvahy je třeba zahrnout charakter vstupu, podle něj zvolit konkrétní metody a jejich optimální zařazení. Vždy je třeba sledovat výstup každé operace a její parametry citlivě nastavit. Velikou roli zde hraje lidská zkušenost.

2.3.3 Předzpracování obrazu

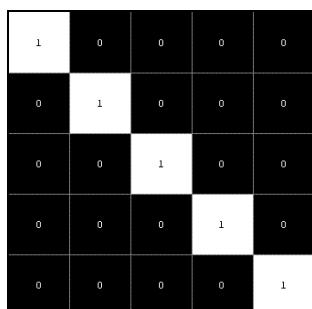
Předzpracování obrazu zařazujeme vždy za operaci získání dat. Důvodem je fakt, že proces zachytávání scény zanáší do obrazu nežádoucí rušivé vlivy. Příkladem je nesprávný úhel snímání a následná potřeba geometrické transformace (zvětšení, rotace, posuv), nebo nerovnoměrné nasvícení, které vyžaduje lokální zpracování (např. rozdělením obrazu do bloků a jejich zpracování zvlášť). Šum v obraze je možné eliminovat filtrací, úprava jasu je možná pomocí vyrovnání histogramu, převodní funkcí lze zvýšit kontrast a podobně. Zajímavé jsou tzv. *morfologické operace*. Jde o sadu postupů, které operují nad tvary objektů. Hlavním principem je aplikace tzv. *strukturního elementu*. Základními morfologickými operacemi je *dilatace* a *eroze*. *Strukturní element* modifikuje vstupní obraz tak, že na výstupu získáváme obraz stejné velikosti jako je velikost obrazu vstupního, přičemž každý *pixel* je výsledkem srovnání tohoto *pixelu* s jeho okolím. Velikost a tvar okolí definuje *strukturní element*. Typ komparace definuje zvolená operace. Určením tvaru a velikosti *strukturního elementu* je možné vytvořit morfologickou operaci citlivou na specifický tvar ve vstupním obraze.

Dilatace

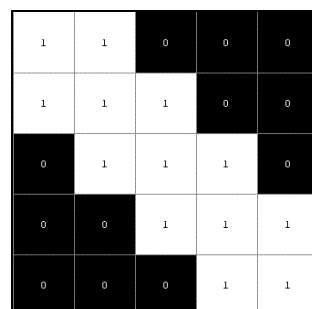
Během dilatace je každému pixelu přiřazena maximální jasová hodnota z jeho okolí. Okolí definuje přiložený strukturní element. Při digitálním zpracování obrazu jsou strukturní elementy typicky reprezentované maticemi s prvky hodnot 0 a 1. Přičemž hodnoty 1 a jejich uspořádání v matici znamená tvar strukturního elementu. Důležité je také definovat střed elementu, který je při operaci pokládán postupně na všechny vstupní pixely. Dilatace zvětšuje tloušťku objektů tím, že rozšiřuje jejich hranice, a zakrývá nespojitosti uvnitř objektů. Touto operací lze např. doplnit chybějící pixely v linii, nebo zakrýt "díry" a úzké štěrbiny mezi objekty. Příklad dilatace binárního obrazu strukturním prvkem je na Obr. 13.



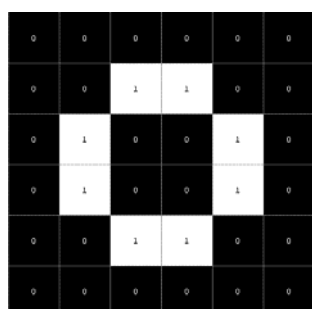
(a)



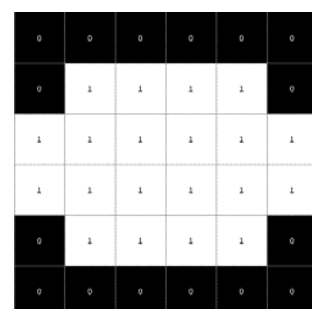
(b₁)



(c₁)



(b₂)

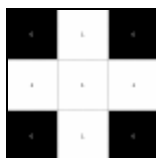


(c₂)

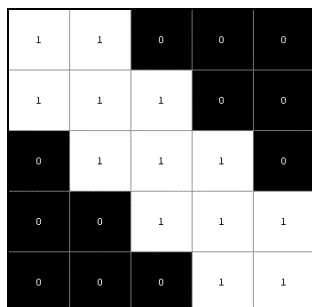
Obr. 13: Dilatace nad binárním obrazem; (a) strukturní element, (b₁),(b₂) originální obraz, (c₁), (c₂) výstupní obraz po dilataci originálu.

Eroze

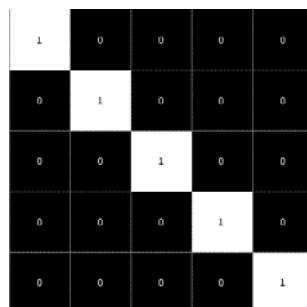
Operace eroze smršťuje a zeštíhluje objekty. Stejně jako dilatace je eroze kontrolována pomocí strukturního elementu. Eroze je opačnou operací k dilataci. Jas každého výstupního pixelu je minimální hodnotou jeho okolí ve vstupním obraze. Výstupní obraz binárního vstupu obsahuje hodnoty 1 jen v místech počátku strukturního elementu, kde tento element zcela překrývá popředí originálního obrazu.



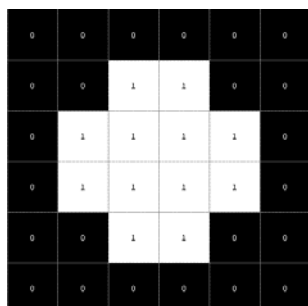
(a)



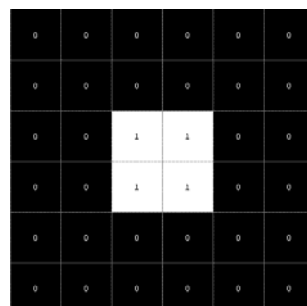
(b₁)



(c₁)



(b₂)



(c₂)

Obr. 14: Eroze nad binárním obrazem; (a) strukturní element, (b₁), (b₂) originální obraz, (c₁), (c₂) výstupní obraz po erozi originálu.

Obě uvedené operace lze také různě kombinovat. Mezi nejznámější komplexní operace složené z těchto dvou základních je *otevření* (*opening*), *zavření* (*closing*) a *top-hat* transformace. Tyto operace lze využít k zvýraznění nebo potlačení specifických rysů v obraze.

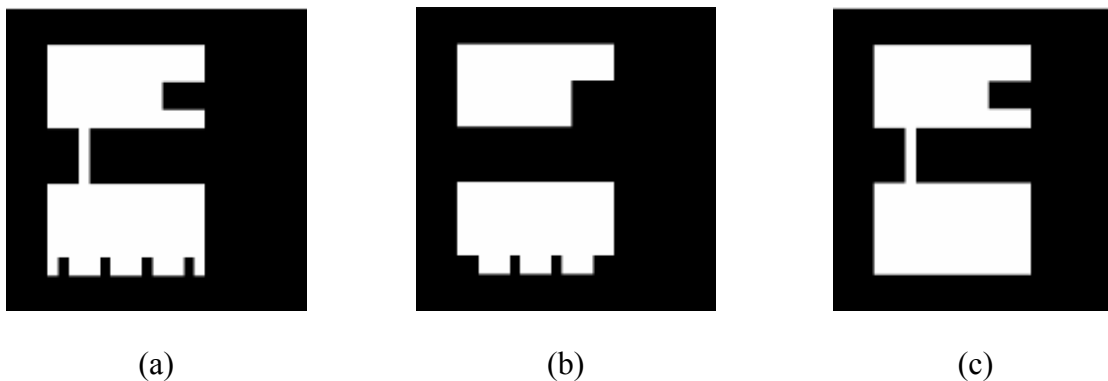
Otevření

Morfologické *otevření* je *eroze* následovaná *dilatací*. *Otevření* odstraňuje oblasti, do kterých se celý strukturní element nevejde, vyhlazuje obrysy, přetrhává úzká spojení mezi objekty a odstraňuje jejich vyčnívající části. Prakticky se z obrazu ve fázi *eroze* odstraní příliš tenké oblasti a velké objekty se zeštíhlí. Strukturní element v této fázi hraje roli sondy, která do další operace propustí jen dostatečně výrazné objekty. Následnou *dilatací* jsou zachované

objekty navraceny do původního stavu před *erozí*, jelikož k oběma operacím je použit shodný strukturní element.

Zavření

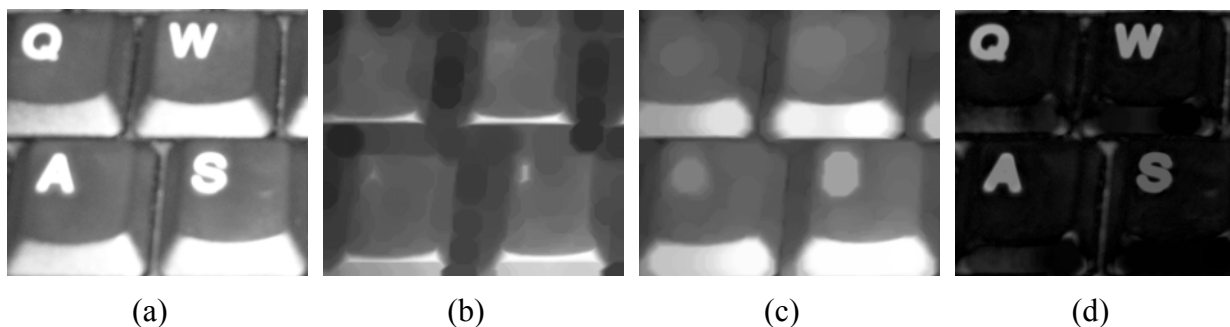
Morfologické *zavření* je *dilatace* následovaná *erozí*. Stejně jako *otevření* se morfologické *zavření* snaží vyhladit obrysy objektů. Oproti *otevření* ale vyplňuje zúžená místa, vyplňuje dlouhé štěrbiny a vyplňuje mezery menší než je strukturní element.



Obr. 15: Morfologie otevření a zavření; (a) originální obraz, (b) výstupní obraz po operaci morfologické otevření, (c) výstupní obraz po morfologické operaci zavření.

Top-Hat transformace

Top-Hat operace odečítá morfologicky otevřený obraz od obrazu originálního. Je dobře použitelná na obraz s nestejnoměrným osvětlením a na vyhledání míst s vysokou intenzitou jasu – zvýšení odstupu od pozadí. Protože jde o operaci složenou ze tří elementárních kroků, je poměrně náročná na procesorový čas. Uvedené příklady ilustrují operace nad binárním obrazem. Lze je však využít i nad obrazem odstínů šedé. Na obrázku jsou vidět postupné kroky v rámci *top-hat* operace během které byl aplikován *strukturní element* ve tvaru kruhu o průměru 25 obrazových bodů.



Obr. 16: Operace top-hat; (a) originální obraz, (b) eroze obrazu (a), (c) dilatace obrazu (b), (d) rozdíl originálního obrazu a obrazu (c).

2.3.4 Analýza dvourozměrného obrazu v prostorové oblasti

Analýza dvourozměrného obrazu nezahrnuje trojrozměrný charakter scény. Je proto vhodná v případech, že jej můžeme v dané aplikaci s jistotou zanedbat.

Prvním krokem je *segmentace*, díky které vyzdvihneme z obrazu zajímavé objekty. To lze provést např. prahovou funkcí, která rozlišením odlišných jasových úrovní rozdělí obraz na pozadí a objekty.

Druhým krokem je kvalitativní popis oddělených objektů. V černobílém obraze získaném *segmentací* lze určovat celou řadu vlastností. Příkladem je počet nalezených objektů, plocha objektu, výška, šířka, orientace, tvar a další. Tyto informace jsou uloženy do vhodného datového typu, odkud lze k vlastnostem konkrétního objektu pohodlně přistupovat.

Úlohou třetího kroku je klasifikace objektů. To znamená rozdělení získaných objektů porovnáním s předdefinovanými vzory do příslušných tříd. Složitost klasifikace je závislá na vzájemné podobnosti objektů v obraze. Snadno rozlišitelné vlastnosti objektů vedou k jednodušším kritériím, zatímco komplexnější scéna žádá sofistikovaný rozhodovací a třídící algoritmus, který využívá např. vah jednotlivých parametrů objektu.

Uvedený postup - segmentace, popisu objektu a roztřídění podle znaků - je možné dále analyzovat, najít kritéria hodnotící pravděpodobnost úspěchu provedených operací a případně změnit jejich parametry pro další snímek, nebo použít na stávající data znovu. Tyto operace jsou zároveň operace nad jedním snímkem. To znamená v jeden časový okamžik dynamického procesu. Algoritmus tak nepočítá s časovým vývojem. Pokud je nutné znát předchozí stav scény již během analýzy obrazu, je třeba použít složitější postup zapojující paměť odpovídající délky.

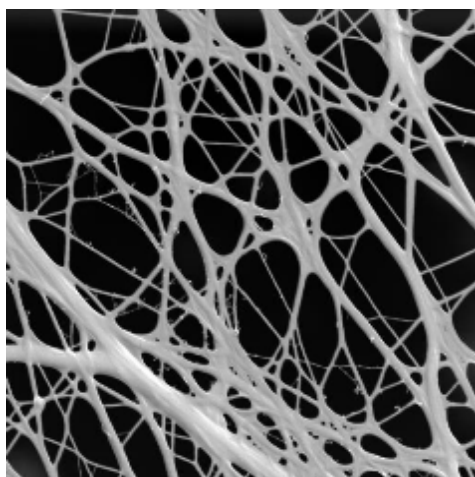
Segmentace

Segmentace identifikuje v obraze oblasti se shodnou vlastností. V šedotónovém obraze mluvíme o oblastech s danou jasovou úrovní. Například pokud je pozadí scény tmavé a okolní osvětlení způsobuje odlesky na objektech, lze tyto objekty od pozadí rozlišit právě odstupem jejich jasu od jasu pozadí. *Segmentace* pak znamená úlohu rozdělení tmavé oblasti do třídy "pozadí" a světlé oblasti do třídy "objekt". Vhodným postupem pro tento účel je metoda *prahování*.

Prahování

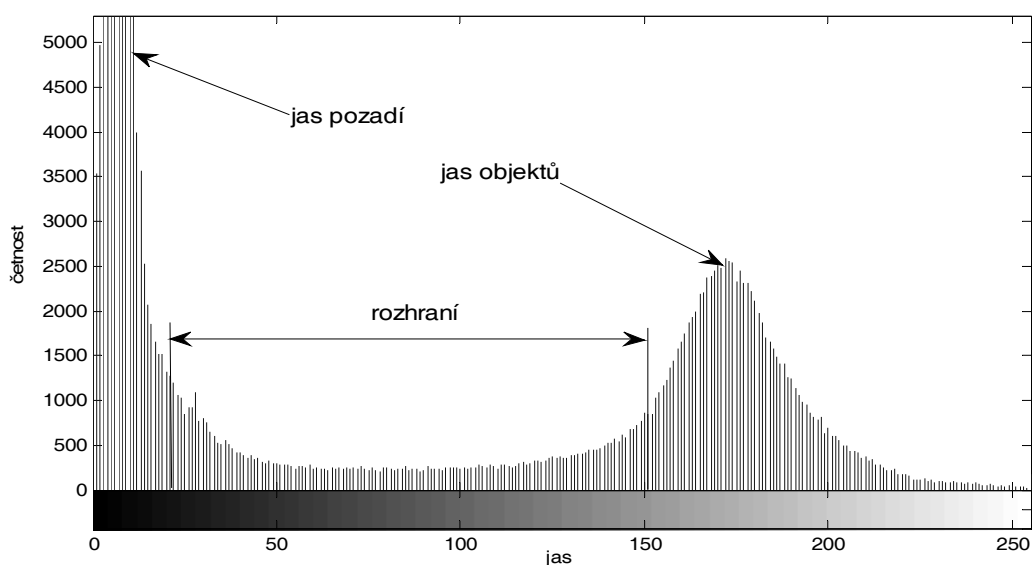
Princip *segmentace* pomocí prahování je velmi jednoduchý. V obraze reprezentovaném odstíny šedé se hodnota jasu každého obrazového bodu porovnává relačním operátorem s *prahovou hodnotou*. Podle výsledku porovnání jsou body s jasovou složkou menší, než je hodnota prahová, zařazeny do třídy "pozadí". Body jasnější než práh spadají do třídy "objekt". Vstupem operace je šedotónový obraz s 256 stupni šedé. Výstupem je obraz binární – hodnoty jasu 1 (objekt), 0 (pozadí) – neboli černobílý obraz.

Osvětlení se v jednotlivých částech obrazu může značně lišit. Pak je použití jednotného prahu na celý snímek nevhodné. Řešením je rozdělení obrazu na více bloků a jejich zpracování zvlášť. Prahování je účinný způsob segmentace obrazu, který charakterizuje stejnoměrné pozadí s dostatečným jasovým odstupem od objektů. Výpočetně je tato metoda nenáročná a velmi rychlá. Rychlost zpracování závisí lineárně s počtem obrazových bodů.



Obr. 17: Obraz reprezentovaný jasovými složkami obrazových bodů.

Problémem této metody je nalezení vhodné prahové hodnoty. Obraz, ve kterém mají objekty zřetelný odstup jasu od jasu pozadí, má *bimodální histogram* úrovně šedé. Dvě maxima v histogramu označují jasové složky pozadí a jasové složky objektů. Interval minima hodnot odpovídá rozhraní mezi nimi. Jednotlivé oblasti jsou tím čitelnější, čím nižší je přítomnost šumu v obraze. Prahovou hodnotu je dobré volit právě v tomto intervalu. Strmost vrcholů vyjadřuje ostrost přechodu mezi pozadím a objektem na jejich hranici. Plochý přechod dovoluje široký interval pro volbu prahové hodnoty. Pak je zřejmé, že tato volba má vliv na plochu získaných objektů. Vyšší práh způsobí zúžení objektů, nižší práh naopak jejich rozšíření. Příklad šedotónového obrazu je uveden na Obr. 17. K němu příslušný histogram je vidět na Obr. 18.



Obr. 18: Histogram jasových složek Obr. 17.

Klasifikace

Objekty oddělené od okolí pomocí segmentace je vhodné pro další zpracování popsat. Popis objektu provádíme na základě charakteristických vlastností. Množina zjištěných vlastností pro daný objekt představuje množinu příznaků, podle kterých můžeme objekty dále logicky zařazovat do předem zvolených tříd. Obecným postupem klasifikace je tedy *nalezení vhodných příznaků* objektu, volba způsobu jejich měření a *konstrukce klasifikátoru*, který objekty na základě příznaků roztřídí. K nalezení správných vlastností neexistuje univerzální metoda. Jde spíše o proces intuitivní. Obecně lze ale říci, že jednotlivé vlastnosti by měly být dostatečně zřetelné a vzájemně odlišitelné s vysokou spolehlivostí. V neposlední řadě, s ohledem na rychlost zpracování, by měla být množina stanovovaných vlastností co možná nejmenší. Pokud se např. budeme snažit v obraze oddělit objekty podle velikosti, pak by měření jejich barvy bylo zcela zbytečné, protože klasifikátor by na základě této informace žádného rozhodnutí schopen nebyl.

Příznaky objektů v černobílém obraze jsou geometrické rozměry – plocha, šířka, výška, orientace, protáhlost, opsaný čtverec s minimální plochou a další. Plocha objektu v binárním obraze je součtem pixelů objektu. Pro stanovení hraničního čtverce objektu počítáme pouze dva pixely – levý horní a pravý dolní. Výška a šířka spočtené z hran opsaného čtverce mohou být zrádné. Pokud se v obraze nachází protáhlý objekt orientovaný v úhlu $\frac{\pi}{4}$ vůči horizontální ose, pak jej pomocí šířky a výšky detekujeme jako čtverec. Štíhlost objektu,

definovaná jako poměr výšky a šířky, je tímto způsobem stanovena jako 1, což neodpovídá skutečnosti. Tento příklad poukazuje na potřebu dalšího příznaku – orientace objektu.

Nejjednodušší klasifikátor objektů redukuje klasifikační rozhodnutí na prahová pravidla. V případě že hodnoty vektoru příznaků spadají mezi stanovené hranice, pak je předmět zařazen do příslušné třídy. Přičemž každá množina prahových hodnot přísluší samostatné třídě. Při konstrukci klasifikátoru jsou většinou prahové hodnoty nastaveny intuitivně a jejich správné odladění se provádí pomocí trénování klasifikátoru množinou známých objektů.

Výstupem klasifikace již není obraz, ale množina objektů popsaných příznaky a rozdělených do příslušných tříd. Klasifikace je posledním článkem v řetězci zpracování obrazu. Se získanými údaji je možné pracovat jinými postupy, než je zpracování obrazu – např. statistickým zpracováním získaného souboru dat.

2.3.5 Fourierova analýza

Fourierova analýza dovoluje nahlédnout na periodicity v předloženém signálu jeho rozložením do lineární kombinace sinusoid o různých frekvencích. Matlab používá k stanovení koeficientů Fourierovy transformace (FT) vzorkovaných diskretních dat výpočet Diskretní Fourierovy transformace (DFT) algoritmu Rychlé Fourierovy transformace (FFT). DFT je pro vektor $x(n)$ definována:

$$X_{(k)} = \sum_{n=1}^N x_{(n)} e^{-j2\pi(k-1)\left(\frac{n-1}{N}\right)}, \quad 1 \leq k \leq N \quad (24)$$

Pro vzorkovaný signál platí tzv. *Nyquistova podmínka*, neboli *vzorkovací věta*, která říká toto: Pokud signál $x_{(t)}$, spojitý v čase, obsahuje pouze frekvenční složky s frekvencemi menšími, než f_{max} , pak je veškerá informace o tomto signálu obsažena v hodnotách $x_{(nT)}$, jestliže vzorkovací frekvence $f_{vz} = 1/T$ je větší než $2f_{max}$. Vzorkovací frekvence musí být tedy větší, než je nejvyšší zastoupená frekvence ve spojitém signálu. Tato podmínka je např. příčinou omezení frekvenčního rozsahu převodníku (viz kapitola Registrace napětí zdroje).

Výsledkem FFT nad vektorem reálných hodnot je vektor stejné délky hodnot komplexních. První prvek, $X_{(1)}$ tohoto vektoru, je součtem všech prvků vektoru $x(n)$.

Výkonová spektrální hustota je definována:

$$P_{(k)} = |X_{(k)}|^2 \quad (25)$$

3 Experimentální část

V experimentální části byl proces elektrostatického zvlákňování z lištové elektrody sledován pomocí videokamery. Pořízené záběry byly zpracovány pomocí analýzy digitálního obrazu. Za tímto účelem bylo třeba vyvinout komplexní softwarové řešení v podobě funkčních algoritmů a grafického uživatelského rozhraní. Programový celek představuje z hlediska časových a intelektuálních nároků rozsáhlejší část než provedení samotného experimentu.

3.1 Chemická část

Veškeré pokusy byly provedeny s bezbarvým vodným roztokem polyvinylalkoholu obchodního názvu Solviol-R o molekulové hmotnosti 60 000g/mol v koncentracích 4-8%hm. Viskozita 4% roztoku byla stanovena 10.4mPa.s. Povrchové napětí roztoků se pohybovalo v okolí 50mN/m a bylo upravováno přidávkem n-butylalkoholu (1-5%hm.), který přispívá k stabilitě trysek. Okolní podmínky byly tyto: teplota 21°C, relativní vlhkost 45%.

3.2 Fyzikální část

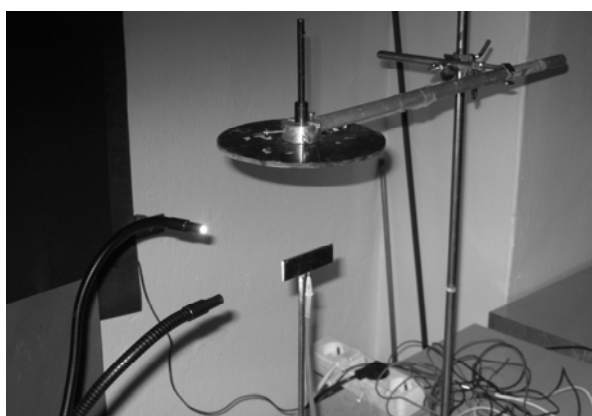
Z hlediska použitých funkčních prvků je vhodné experiment rozdělit na oblast měřenou, kterou představuje zvlákňovací sestava a na oblast měřících zařízení.

3.2.1 Zvlákňovací sestava

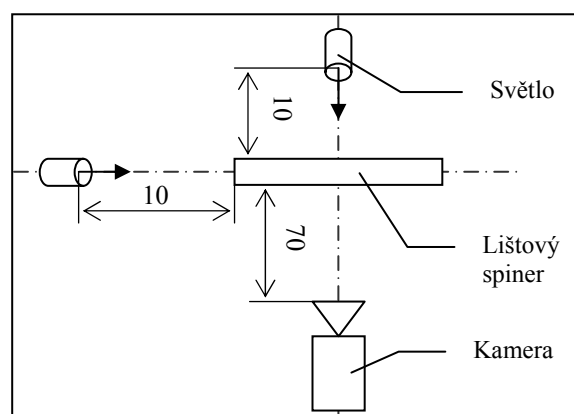
Experimentální zařízení sestává z uzemněného *lištového spinneru* orientovaného tak, aby volná hladina polymeru směřovala směrem vzhůru ke *kolektoru*. *Kolektor* je železná deska kruhového tvaru o průměru 150 mm. Vzdálenost mezi kolektorem a hladinou na spinneru byla během experimentů v rozsahu 70-100mm. Na *kolektor* je přivedeno napětí z vysokonapěťového zdroje Glassman Watt High Voltage DC Power Supply, model PS/ER50N06.0-22. Výstupní napětí je možné mechanicky potenciometrem, ale také elektronicky na zadních vstupech regulovat v rozsahu 0-50kV. Zásoba polymeru ve spinneru byla udržována pomocí injekční pumpy Polymed IPC 2050 a injekční stříkačky Chirana o objemu 20ml. Objemové průtoky se pohybovaly v závislosti na napětí od 7-16ml/hod. Základní sestavu potřebnou k provedení elektrospiningu tak tvoří čtyři prvky: lištový přípravek (spinner), napěťový zdroj, injekční pumpa a kolektor.

3.2.2 Osvětlení scény

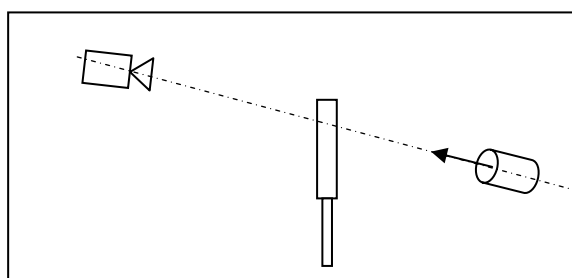
Vznikající trysky jsou s ohledem na roztok bezbarvé a kvůli malému průřezu jsou jen velmi málo patrné. Z tohoto důvodu byla za spinnerem zavěšena černá matná stěna z papíru a úzké okolí nad hladinou polymeru bylo nasvíceno dvěma bodovými reflektory. Jedno ze světél, v pozici vlevo od spinneru, bylo ve výši hladiny orientováno tak, aby paprsek světla směřoval podél roviny trysek a osvětloval je z boku. Druhé světlo, postavené z hlediska videokamery za spinnerem v polovině jeho výšky, je orientováno tak, aby svítilo téměř přímo do středu objektivu kamery. Při správné poloze světla vůči liště a kameře je paprsek spinnerem zakrytý a objektiv kamery zůstává ve stínu. Rozmístění prvků je patrné z Obr. 19. Jako zdroj světla bylo použito zařízení Euromex Illuminator EK-1. V laboratoři jsou všechny jiné zdroje světla potlačeny, aby bylo dosaženo co možná nejvyššího kontrastu mezi vznikající tryskou a pozadím.



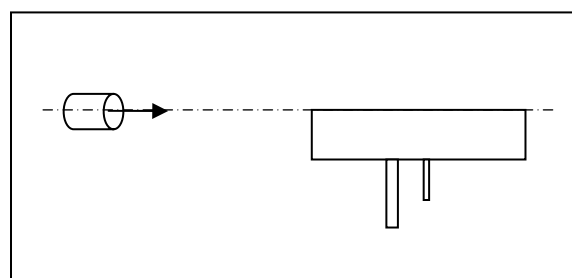
(a)



(b)



(c)

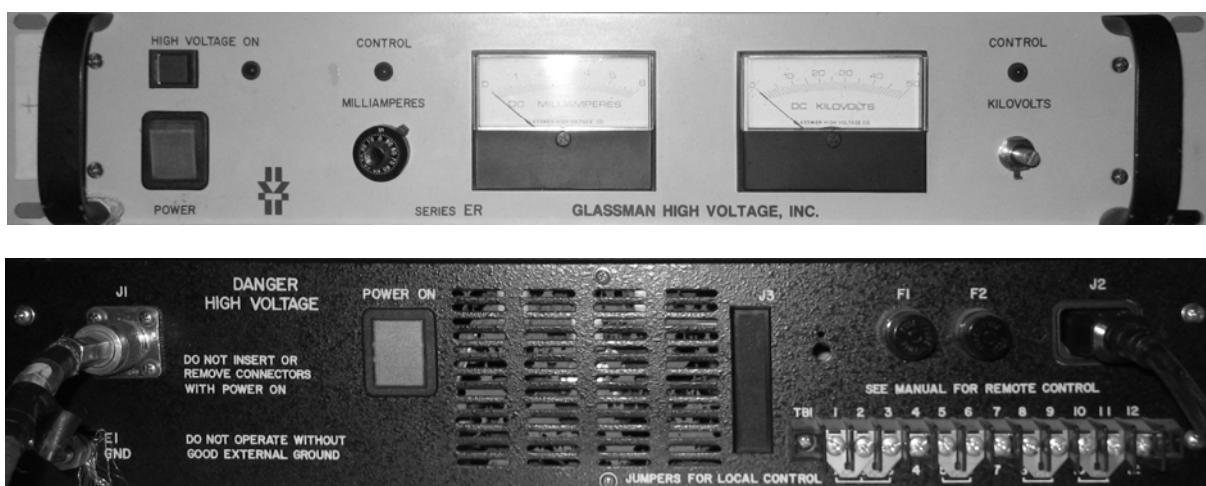


(d)

Obr. 19: Schéma postavení kamery a světél během experimentu. Rozměry jsou v centimetrech; (a) fotografie aparatury, (b) přodorys celé sestavy, (c) poloha zadního světla vůči spinneru a kameře (bokorys), (d) poloha bočního světla vůči spinneru (nárys).

Nutno podotknout, že správné osvětlení je pro zpracování obrazu klíčovou záležitostí. V případě osvětlení trysek jde o úkol poměrně náročný. Proto je nutné přistupovat k osvětlování scény s patřičnou trpělivostí a pečlivostí.

Před scénu byla na stativu umístěna kamera ve vzdálenosti 80cm od spinneru. Aktuální napětí na elektrodách je sledováno pomocí převodníku připojeného na zadní výstupy vysokonapěťového zdroje.

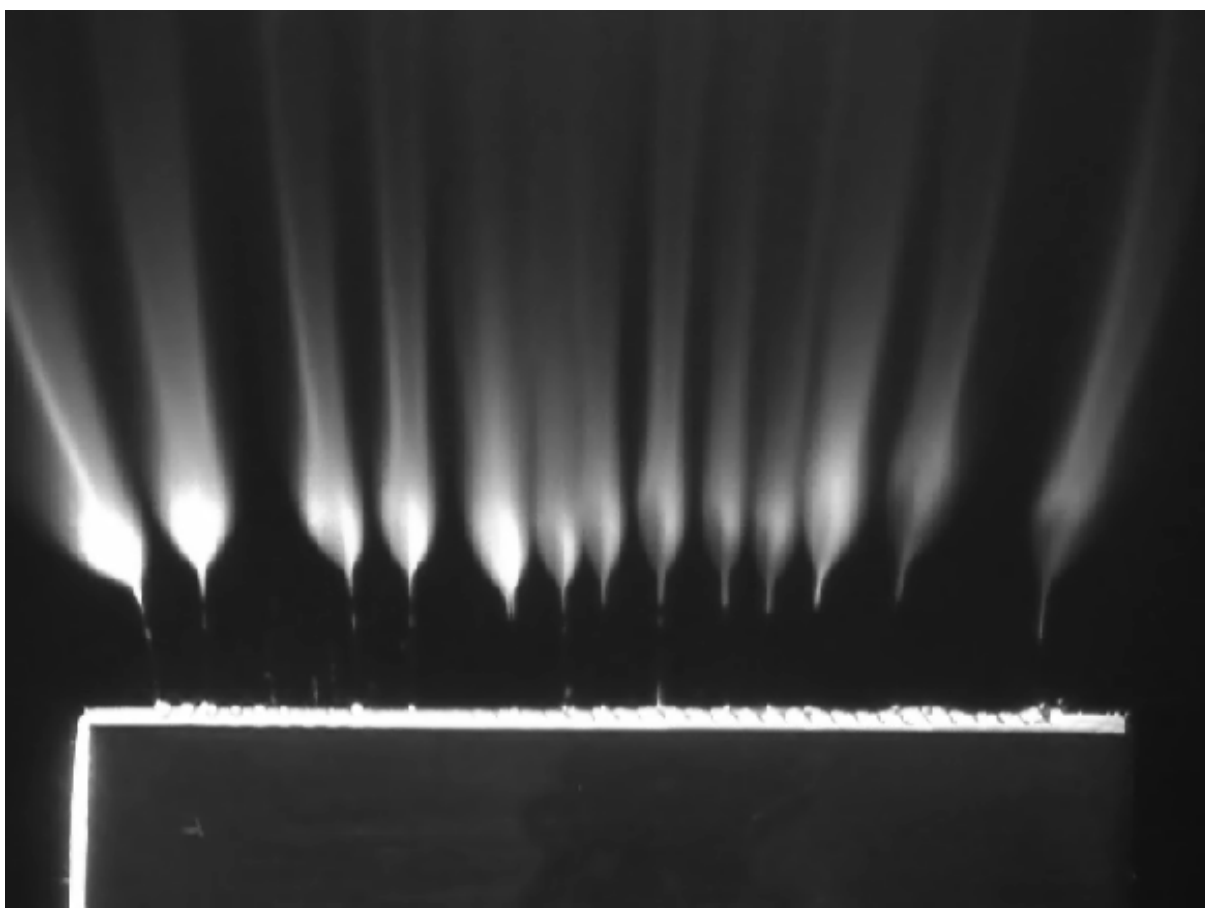


Obr. 20: Fotografie předního a zadního panelu zdroje vysokého napětí.

3.2.3 Prostředky získání dat

Obraz

Pro získání obrazových dat byla použita *miniDV* kamera Panasonic NV-GS120. Aby byla ztráta perspektivy během natáčení minimální, je třeba kameru umístit tak, aby rovina snímaného obrazu byla rovnoběžná s rovinou, kterou vytyčuje stěna spinneru, a zároveň tak, aby se horizontální střed spinneru nacházel ve středu obrazu. Dolní okraj spinneru je umístěn na spodním okraji obrazu. Napravo a nalevo obrazu je vhodné ponechat malou mezeru od bočnic, zjednodušuje se tak automatická detekce polohy hladiny a délka spinneru vhodně slouží jako měřítko pro přepočítání rozměrů v obraze do rozměrů skutečných. Hladina polymeru se při tomto nastavení nachází zhruba v jedné čtvrtině výšky obrazu. Příklad snímku získaného správným postavením kamery a světel je možné vidět na Obr. 21.



Obr. 21: Obrázek získaný při správné poloze světla a videokamery vůči lištovému spinneru.

Kameru je vhodné z bezpečnostních důvodů umístit raději dále od aparatury a pomocí funkce přiblížení získat správný obraz. Během experimentů byl na kameře představen makroobjektiv. Dnešní kamery používají tzv. digitální přiblížení (*zoom*), jeho použití je ovšem třeba naprosto vyloučit, jelikož tato metoda snižuje rozlišení obrazu. V aplikaci, o níž bude psáno dále, je možné zvolit zpravování ze záznamu nebo přímo v reálném čase. Formát ukládání dat *miniDV* má své výhody i nevýhody, celkově je však pro aplikaci nevhodným vstupem hned z několika důvodů. *MiniDV* částečně dodržuje evropský standard *PAL*. Z něho přebírá rozlišení 720x576px a frekvenci 25 snímků za sekundu. Bohužel, kvůli zpětné kompatibilitě s *CRT* monitory je uložený záznam tzv. *prokládaný (interlacing)*. To znamená, že jsou v záznamu přítomné půlsnímky s frekvencí 50Hz složené z lichých a sudých řádků. Zpravování obrazu metodami popsanými v teoretické části této práce vyžaduje, aby na vstupu procesu byl přítomen snímek celý. Před zpracováním záznamu je proto třeba jeho úprava metodou zvanou *deinterlacing*. Zároveň je formát *miniDV* velmi neúsporný. Vteřina záznamu zabere v paměti počítače 3,6MB dat. Práce s takovým množstvím dat je časově náročné a zdržuje vykonávání funkcí programu. Z těchto dvou důvodů byl před vstupem do aplikace videozáznam

komprimován do formátu *WMV9* při datovém toku minimálně 600kb/s a se zapnutou funkcí *deinterlacing*. Ostatní parametry obrazu, jako je rozlišení, zůstaly beze změny.

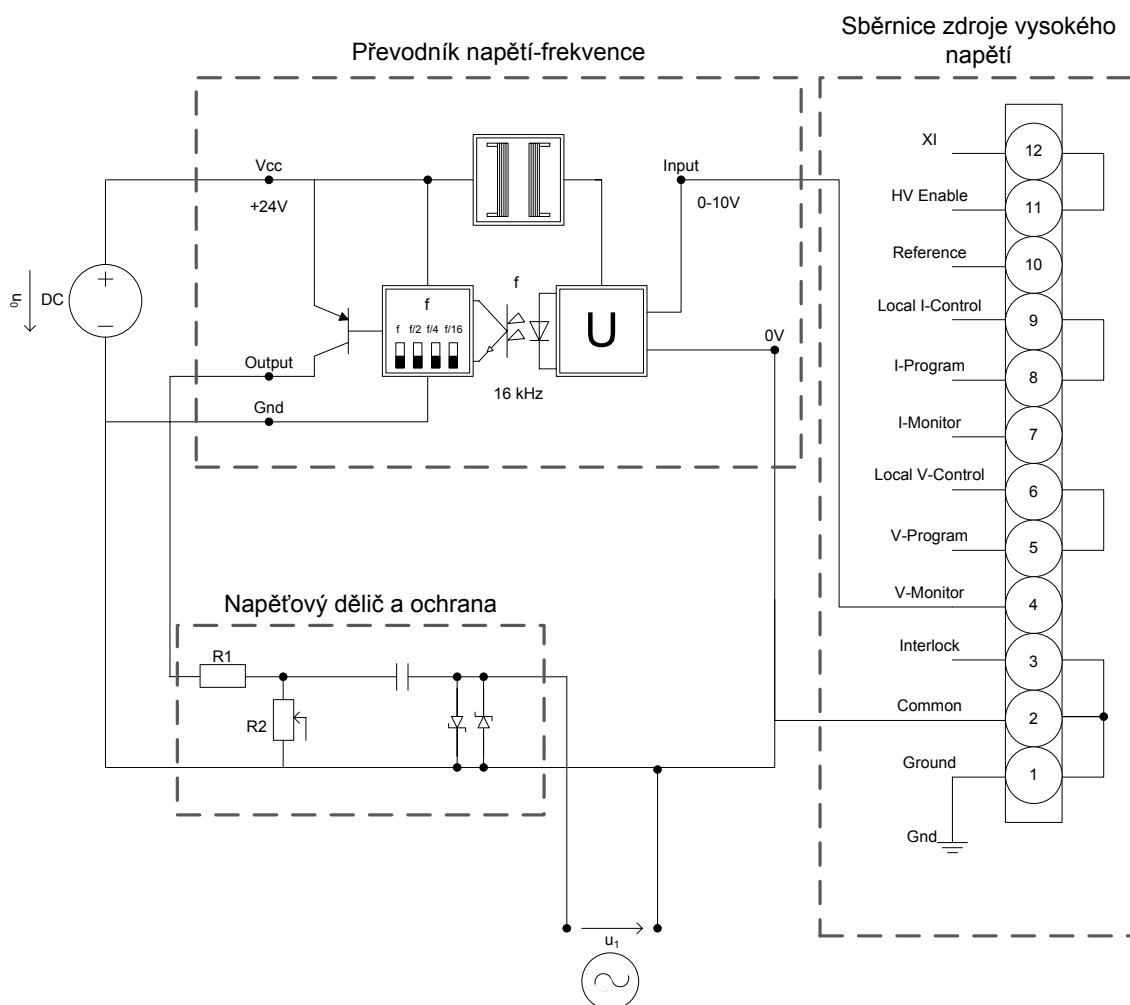
Při zpracování dat v reálném čase jsou snímky sbírány okamžitě přes rozhraní *IEEE 1394*. Počet zpracovaných snímků v jedné vteřině závisí na výkonu použitého PC. Při zpracování obrazu v reálném čase aplikace snímky pro pozdější použití neukládá.

Implementace dvou různých způsobů zpracování experimentu, v reálném čase a ze záznamu, byla původně provedena z obavy před možnou indukcí elektrického náboje v okolí aparatury. Náboj, který se hromadí na všech předmětech, se při uzemnění daného předmětu v krátkém okamžiku vybíjí. Veškeré počítačové komponenty jsou na poškození elektrostatickým výbojem velmi citlivé. Během zpracování videozáznamu je ale počítač bezpečně od aparatury oddělen. K zpracování v reálném čase byl použit stolní počítač. Napěťový zdroj byl připojen na stejný zemnicí vodič jako vodič zdroje PC a k očekávaným problémům zatím nedošlo.

Napětí

Hlavním cílem práce bylo stanovení počtu a vzdálenosti trysek v závislosti na napětí elektrod. Použitý zdroj vysokého napětí je na zadní straně opatřen konektory, které slouží jako analogové rozhraní pro řízení a sledování stavu přístroje. Rozhraní je standardní: napětí v rozsahu 0-10V, proud 0-200mA. K sledování napětí na zdroji byl sestaven převodník vlastní konstrukce. Hlavním důvodem pro stavbu byla myšlenka, že každé dnešní PC obsahuje velmi kvalitní AD převodník v podobě zvukové karty. Matlab dokáže díky *Data Acquisition Toolboxu* přistupovat k jakékoli zvukové kartě v systému Windows, pokud k ní existují ovladače modelu *WDM* – a tato podmínka je splněna v naprosté většině osobních počítačů. Převodník zpracovává na vstupu napětí v rozsahu 0-10V, výstupem je podle konfigurace střídavý signál s amplitudou 200mV a frekvencí 0-16kHz.

Celý převodník sestává z napájecího zdroje, samotného frekvenčního měniče a výstupního obvodu.



Obr. 22: Schéma zapojení převodníku a jeho propojení s rozhraním zdroje vysokého napětí. Náčrtek oblasti frekvenčního měniče je převzato z katalogu firmy Weidmüller.

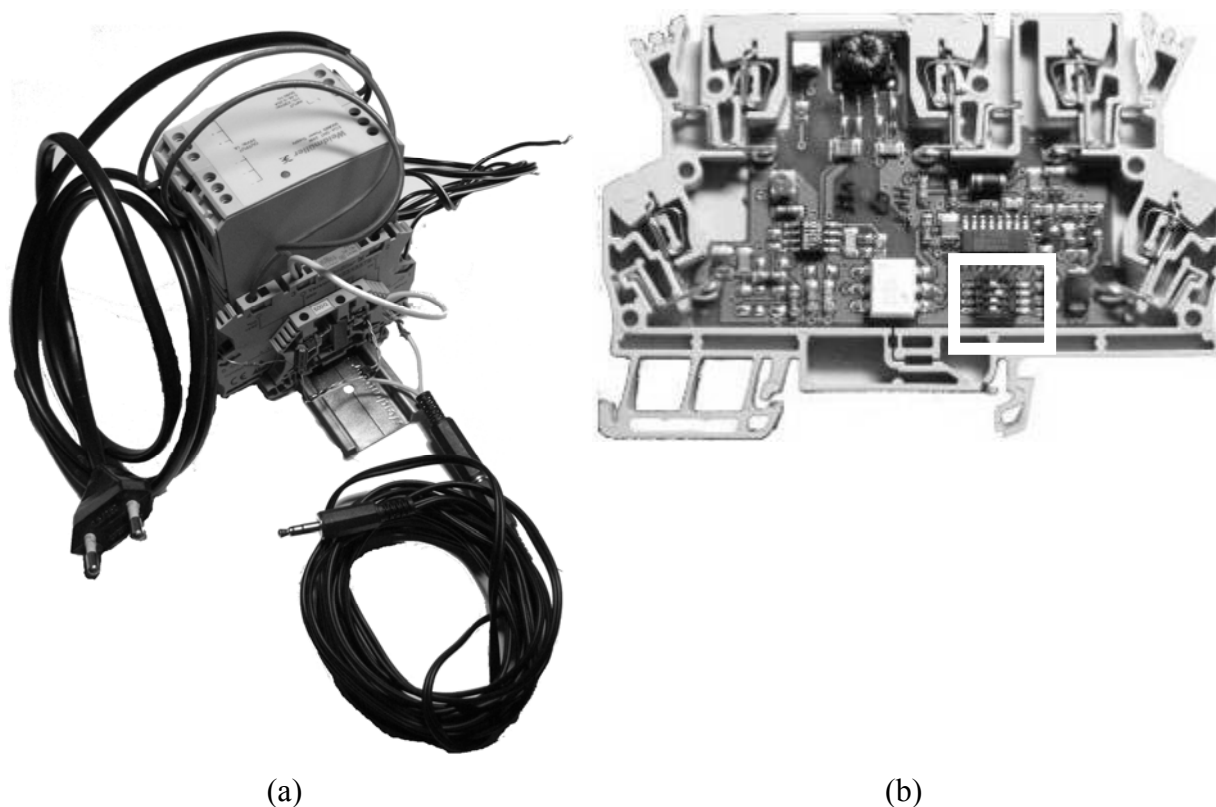
Napájecím zdrojem pro převodník je Weidmüller EG4-SNT Power Supply. Ten mění střídavých 230V z běžné sítě na stejnosměrné napětí 24V.

Funkční jednotkou převodníku je měnič stejnosměrného napětí na frekvenci: Weidmüller MCZ VFC. Sledované vstupní napětí 0-10V slouží jako činitel změny frekvence výstupního napětí takto: 0V odpovídá 0Hz, 10V odpovídá 16kHz. Amplituda napětí na výstupu měniče je $\pm 12V$.

Výstupní obvod převodníku tvoří rezistor, potenciometru, kondenzátoru a dvě rychlé Schottkyho diody. Rezistor $R_1=5,5k\Omega$ slouží společně s potenciometrem $R_2=0-500\Omega$ jako napěťový dělič k redukci $\pm 12V$ na $\pm 200mV$. Kondenzátor $C=4,7\mu F$ slouží k případnému odstranění stejnosměrné složky signálu. Dvě Schottkyho diody zapojené antiparalelně v závěrném směru způsobují svedení napětí na zem v případě, že jeho hodnota přesáhne

300mV. Tvoří tak jednoduchou přepětovou ochranu. Převodník je napojen na piny 2 a 4 vysokonapětového zdroje. Výstupní vodiče převodníku jsou zakončeny standardním 3,5mm stereo konektorem.

Takto upravený signál je možné zapojit do běžné zvukové karty. Zvuková karta mění analogový vstup v jeho digitální reprezentaci. Podobně jako při zachytávání obrazu jde o proces vzorkování a kvantizace spojitě veličiny.



Obr. 23: Převodník použitý pro registraci napětí na elektrodách; (a) sestavený převodník, (b) měnič stejnosměrného napětí na frekvenci. Bílý čtverec vyznačuje přepínače, kterými lze měnit frekvenčního rozsahu převodníku.

Frekvenční rozsah měniče je možné přizpůsobit pomocí čtyř přepínačů na jeho desce. Možné rozsahy jsou 0-1kHz, 0-4kHz, 0-8kHz, 0-16kHz. Rozsah byl při experimentech z realizačních důvodů omezen na 0-8kHz.

Napětí na vysokonapětovém zdroji je tak sledováno pomocí zvukového tónu v oblasti slyšitelné lidským uchem. Této skutečnosti lze s výhodou využít při natáčení záznamu pro pozdější zpracování. Při této metodě je výstup převodníku pomocí standardního 3,5mm stereo konektoru zapojen přímo do mikrofonního vstupu videokamery. Nahraná zvuková stopa poté slouží jako nosič informace o aktuálním napětí ve sledovaném videu.

3.3 Programová část

Tato kapitola se zabývá realizací experimentu z hlediska potřebného softwarového vybavení. Obsahem je krátký popis uživatelského rozhraní. Na principiální úrovni je vysvětlena činnost prováděná v hlavních funkcích aplikace. Zmíněna je též struktura zdrojového kódu.

3.3.1 Volba vývojového prostředí

Reprezentace digitálního obrazu jako matice přímo poukazuje na to, aby se vývojovým prostředím stal systém Matlab. Jde jednak o integrované vývojové prostředí pro technické výpočty rozšiřitelné pomocí tzv. *toolboxů*, jakýchsi tematicky sjednocených knihoven funkcí, zaměřených na právě jednu oblast výpočtů. Zároveň jde o programovací jazyk vysoké úrovně. Matlab, a to je jeho neskutečnou výhodou, používá jako fundamentální datový typ matici. Ostatní programovací jazyky obsahují datové typy skalární. Implementují také typy polí, ovšem pro pohyb po prvcích těchto polí je třeba vždy zapojit hledání pomocí skalární indexace. Tento přístup vede k používání *for* nebo *while* cyklů, jejichž zpracování je nepřehledné, komplikované a hlavně při běhu programu relativně pomalé. Matlab je navržen pro vektorový a maticový počet, a tak lze některé operace v cyklech převést a použít velmi rychlou indexaci pomocí vektorů - tzv. *vektORIZACE kódu*. V programovacích jazycích, jako je C# nebo Java, takové zjednodušení možné není. Jak již bylo řečeno – digitální obraz je reprezentován maticí a výhoda indexování používaného v Matlabu vychází rychle najevo. Jednou z oblastí, na které je systém Matlab dobře připraven je zpracování obrazových dat díky rozšiřujícímu modulu nazvaného *Image Processing Toolbox* (dále jen *IPT*). *IPT* je sada předdefinovaných funkcí nízké a střední úrovně, které jsou použitelné jako základ pro psaní sofistikovanějších algoritmů aplikovaných na konkrétní potřeby. Vyvíjená aplikace byla psaná v prostředí Matlab verze 7.5.0.342 (R2007b) rozšířené mimo jiné o *IPT* verze 6.0. Použita byla i další rozšíření, jejichž seznam je uveden na začátku dokumentu.

3.3.2 Vytvořená aplikace

Při prvních pokusech o registraci polohy trysek během hladinového zvlákňování bylo použito prostředí Simulink, které díky svému grafickému uživatelskému rozhraní dovoluje rychlý návrh systému bez nutnosti ručního psaní programového kódu. V souvislosti s představou možného použití *RealTime Workshop* a *xPC Target*, se tento přístup zdál velmi progresivní. Výsledky, které navržený model poskytoval, byly poměrně dobré, ovšem časem vyšly najevo další požadavky, které v tomto prostředí nebylo možné splnit. Mezi hlavní důvody patří jistá

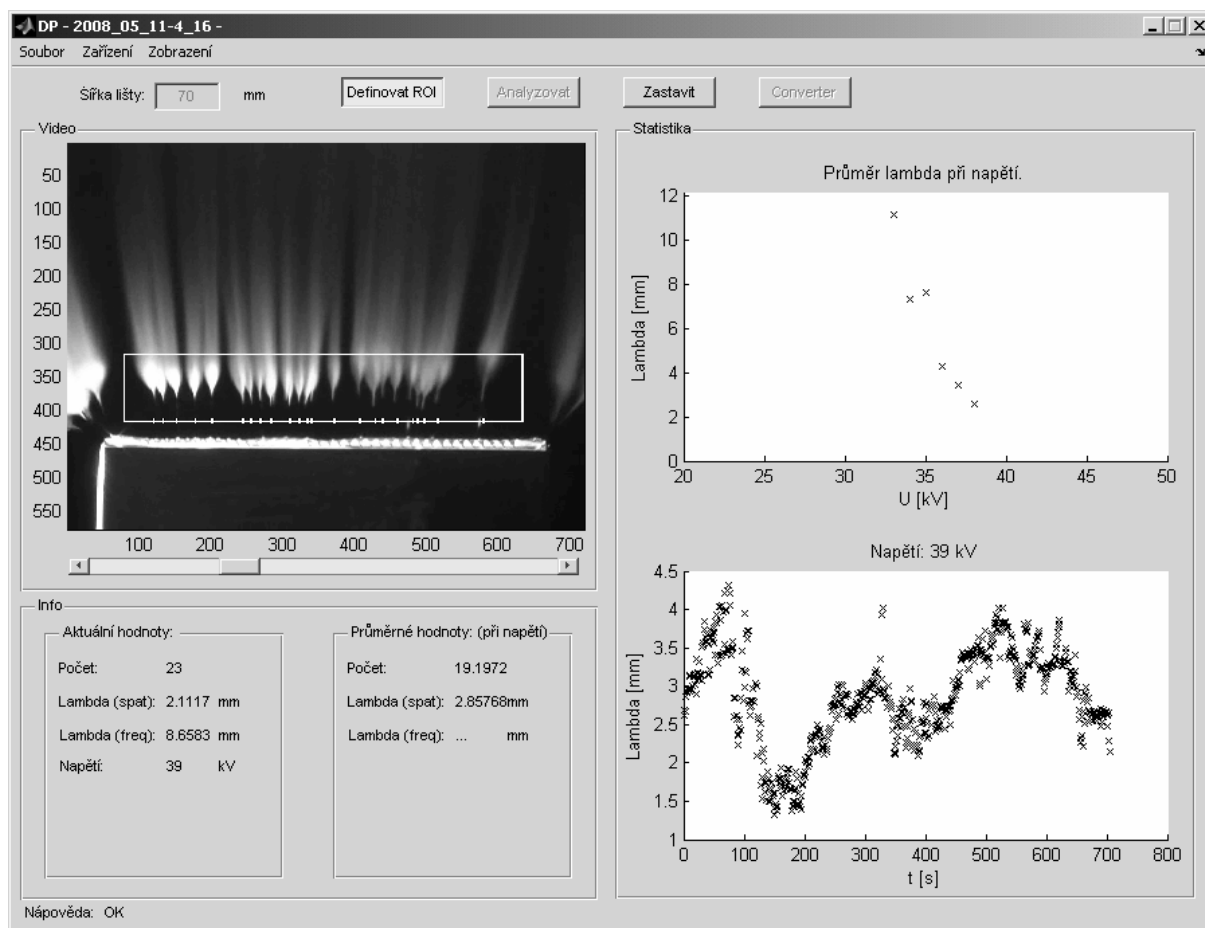
omezení v přístupu k datům a v možnostech manipulace s nimi. Ve chvíli, kdy je třeba získat okamžitý přístup a přehled nad všemi proměnnými, nahlédnout do vnitřku datových struktur, navrhnout vlastní strukturu nebo funkci, je lepší zvolit klasický přístup psaní zdrojového kódu. To je možné i v Simulinku a některé *vlastní funkce* byly takto skutečně realizovány. V souvislosti s plánovaným nasazením aplikace na *xPC Target* bylo nutné psát uživatelské funkce v jazyce C a to je v porovnání s psaním *m-kódu* nadmíru nepohodlné. Bohužel vyšlo najevo, že v *xPC Target* by nebylo možné použít měření napětí zdroje přes zvukovou kartu. Dalším podstatným důvodem pro opuštění koncepce Simulink modelování, byl záměr vybudovat samostatně spustitelnou aplikaci s jednoduchým uživatelským rozhraním, kterou by bylo možné instalovat a spouštět na jakémkoli počítači bez nutnosti předchozí instalace Matlabu. Přestože obecná strategie zpracování obrazu zůstala od počátku vývoje nezměněná, stávající aplikace je kompletně napsaná v *m-kódu*.

Grafické uživatelské rozhraní

Pro pohodlnou a jednoduchou práci jak během experimentu, tak při pozdějším zpracování dat bylo sestaveno grafické uživatelské rozhraní (*GUI, Graphical User Interface*). Při jeho návrhu byl kladen důraz na intuitivní ovládání a dodržení nejvyšší možné podobnosti s jinými běžně používanými aplikacemi. Pro nového uživatele by neměl být problém se rychle zorientovat a v krátkém čase začít aplikaci používat.

Pro vývoj uživatelského rozhraní bylo použito nástroje *GUIDE* obsaženého v základní instalaci Matlabu. Jde o grafické rozhraní pro tvorbu uživatelských aplikací. Návrh okna a ovládacích prvků je realizován v grafickém prostředí. Grafická podoba aplikace a další náležitosti jsou uloženy ve zdrojovém souboru s příponou ".fig". K tomuto souboru přísluší soubor stejného názvu s příponou ".m", který obsahuje logickou část aplikace. Tento soubor je mimo jiné zdrojový kód tzv. *callback* funkcí, tj. obslužných funkcí událostí, ke kterým dochází interakcí uživatele s rozhraním aplikace.

Jádrem aplikace je popis rozhraní v souboru "DP.fig" a příslušný zdrojový kód obsažený v souboru "DP.m". Funkce DP je vstupní funkcí programu a musí být definována jako hlavní při kompilaci programu.



Obr. 24: Hlavní okno aplikace. V horní části panel nabídek a tlačítka pro ovládání záznamu. Vlevo náhled na zpracovávaný obraz, v ostatních částech okna jsou textově a graficky znázorněny průběžné statistické údaje. V dolním pruhu se zobrazuje okamžitá nápověda k prováděným činnostem.

Rozvržení zdrojového kódu

Způsob psaní zdrojového kódu v Matlabu dovoluje oddělení grafické části aplikace od její části logické. Zároveň lze pro zlepšení přehlednosti kódu oddělit jednotlivé funkce do samostatných souborů. Zdrojový kód aplikace je tímto způsobem strukturován a seznam nejdůležitějších funkcí je uveden níže.

AdjustVoltage.fig, Samostatné okno, které je použito pro kalibraci napětí zdroje a frekvenční odezvy převodníku.

AdjustVoltage.m

AnalyzeImage.m Tato funkce zpracovává, analyzuje a klasifikuje objekty ve vstupním obraze. Implementuje nejdůležitější algoritmus v celém systému.

DP.fig, DP.m

Soubory hlavního okna aplikace. Jejich úkolem je vykreslování

grafických objektů. Zároveň také sdružují všechny ostatní funkce. Udržují globální paměť programu za běhu, volají ostatní funkce ve správném pořadí a spravují jejich návratové hodnoty.

<i>DevicePopUp.fig,</i> <i>DevicePopUp.m</i>	Samostatné okno, které dovoluje nastavit zdroj videa před měřením v reálném čase. Zdrojem videa může být jakákoli videokamera, nebo webová kamera připojená přes rozhraní IEEE 1394 nebo USB 2.0.
<i>GetROI.m</i>	Velice důležitá funkce, která zprostředkovává automatické nalezení zajímavé oblasti ve vstupním obraze. Zajímavou oblast představuje šířka spinneru, hladina polymeru a část ve které se během elektrospiningu objevují trysky. Výstup této funkce je jedním ze vstupních argumentů funkce <i>AnalyzeImage</i> .
<i>NewTrialPopUp.fig,</i> <i>NewTrialPopUp.m</i>	Samostatné okno, které zprostředkovává založení nového experimentu a poskytuje přístup k zadávání informací o podmínkách během jeho realizace.
<i>settings.mat</i>	Datová struktura uchovávající nastavení aplikace v době od vypnutí aplikace do jejího nového spuštění. Zde je např. uložena převodní funkce získaná po kalibraci zdroje napětí.
<i>Statistics.fig,</i> <i>Statistics.m</i>	Samostatné okno dostupné po ukončení měření. Dovoluje statistický náhled na data získaná během experimentu. Svou důležitostí odpovídá funkci <i>AnalyzeImage.m</i>

Hlavní funkce

Aplikace sestává z 281 funkcí uložených v 34 samostatných souborech. Níže jsou uvedeny principy užití v nejdůležitějších funkcích. Text neobsahuje zdrojový kód, zřídka jsou uvedeny pouze názvy užitých vnitřních funkcí Matlabu.

Nalezení měřítka a hladiny polymeru

Přesto že videokamera zachycuje poměrně značnou část scény, z analytického hlediska je zajímavý pouze úzký pás těsně nad hladinou polymeru. Zde dochází k samoorganizaci trysek, ze kterých jsou dále formována nanovlákná. Tryska je v obraze poměrně dobře rozeznatelná od pozadí díky odlesku světla z reflektorů. Zhruba jeden centimetr nad hladinou polymeru se tryska rozštěpí ve vějířovitý útvar. Jde o oblast samotných nanovláken. Vlákenné proudy ze sousedních trysek se překrývají a tvoří nedefinovatelné shluky. Oblast nad rozštěpením trysky

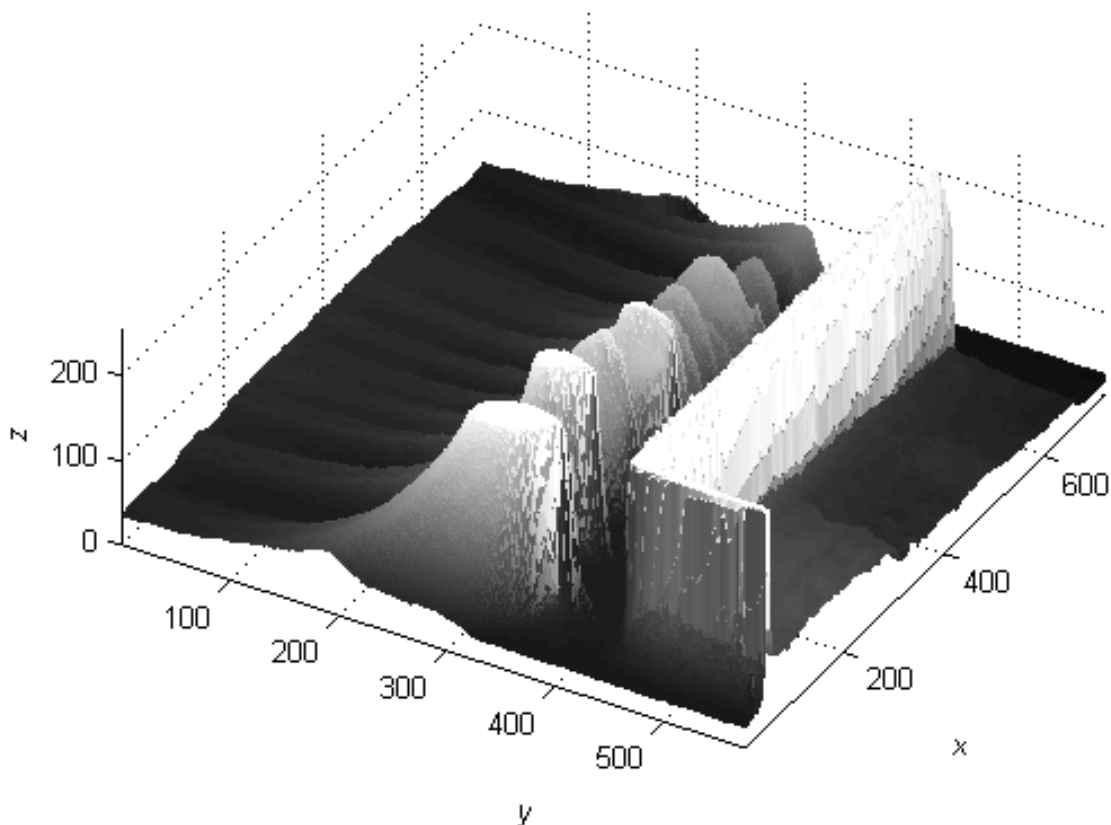
již není z hlediska nalezení polohy trysek zajímavá a její zpracování by zbytečně zvyšovalo nároky na objem přenášených dat a výpočetní čas. Proto se z vstupního obrazu zpracovává pouze důležitý výřez. Definici zajímavé oblasti v obraze může provést uživatel ručně pomocí myši, avšak v základním nastavení dojde k pokusu o nalezení správného výřezu pomocí funkce *GetROI*.

Poznámka: ROI je zkratka pro Region Of Interest, oblast zájmu.

Vstupními argumenty funkce jsou obraz z videokamery a hodnota skutečné šíře spinneru. Tu je třeba v aplikaci zadat ručně před zahájením experimentu. Při zpracování obrazu zjišťujeme vzájemnou polohu objektů v rozměru obrazových bodů. Pro přepočítání do skutečných fyzických proporcí musíme znát, jakou délku jeden obrazový bod ve scéně představuje. K tomu slouží měřítko. Měřítkem může být jakýkoli objekt v obraze, jehož rozměr je předem známý.

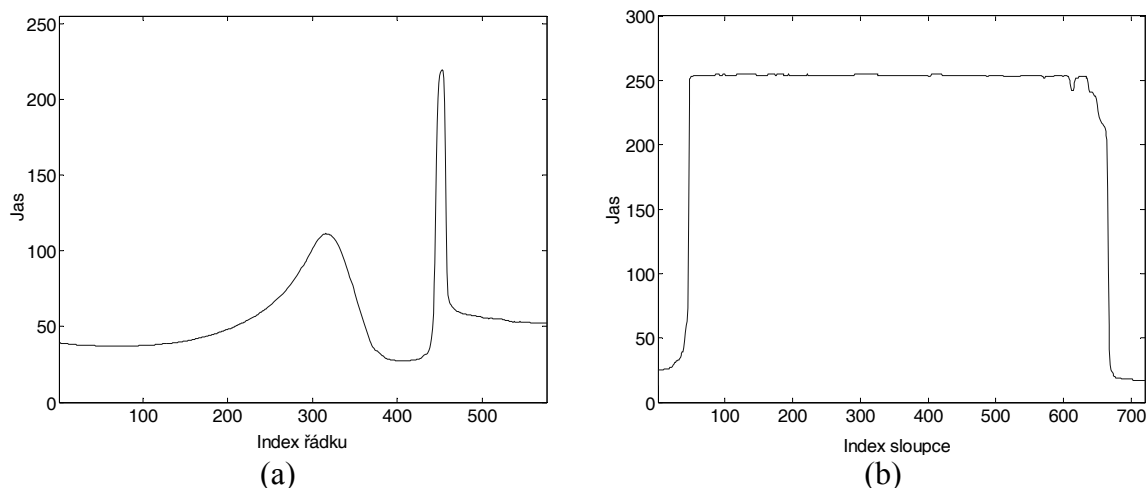
V teoretické části byl digitální obraz definován jako funkce dvou proměnných: $z = f(x, y)$. Skloněním pohledu na obraz tak, aby bylo možné jasové složky vynést na osu z , obdržíme trojrozměrný graf funkce dvou proměnných, jak je zřetelné z Obr. 25.

Funkce *GetROI* využívá pohledu zleva doprava (podél osy x) a zespodu nahoru (podél osy y) v Obr. 25 k nalezení horizontálního a vertikálního profilu jasových úrovní. Oba profily jsou znázorněny na Obr. 26. V obraze o rozlišení 720x576px představuje každý z 576 řádků vektor 720 prvků. Každý jeden prvek vektoru představuje jasovou úroveň v obraze. Spočteme-li průměrnou hodnotu všech prvků tohoto vektoru, získáme průměrnou jasovou úroveň v daném řádku. Vynesením průměrné hodnoty jasu pro každý řádek do grafu je získán vertikální profil.



Obr. 25: *Prostorové zobrazení jasových úrovní z Obr. 21. Hodnota jasu v každém bodě je promítnuta do osy z. Pohled je umístěn 30° nad rovinu x, y a otočený podle osy z o 38° od vodorovné osy x.*

Při dobrém nasvícení scény se horní lišta spinneru ve vertikálním profilu projeví jako místo s nejvyšší průměrnou jasovou úrovní. Tato skutečnost je zřejmá již z toho, že horní lišta během natáčení scény prochází téměř po celé šíři obrazu a zároveň je po délce osvětlena reflektorem zleva. Maximum vertikálního profilu slouží jako odhad pozice horní lišty spinneru. Toto maximum zároveň dělí obraz na dvě poloviny – oblast trysek 'vlevo' a oblast spinneru 'vpravo'. Pro stanovení měřítka je třeba nalézt v obraze šíři spinneru. Princip stanovení je shodný se stanovením vertikálního profilu. Úloha je pouze otočena o 90°. Horizontální profil je vypočten pro řádek, který označuje hladinu polymeru – maximum vertikálního profilu. Jde v podstatě o zobrazení všech prvků nejsvětlejšího řádku v obraze. Na horizontálním profilu jsou jasně viditelná místa rychlého náběhu a poklesu jasové úrovně. V derivaci horizontálního profilu je levá bočnice maximem, pravá bočnice minimem. Vzdálenost těchto extrémů je odhadem šířky spinneru. Horizontální profil je 'vyhlazen' pomocí jednorozměrného mediánového filtru.



Obr. 26: Vertikální a horizontální profil získaný z Obr. 25; (a) vertikální profil, (b) horizontální profil.

Výstupními hodnotami funkce GetROI jsou souřadnice zajímavé oblasti a konstanta pro přepočet vzdálenosti v pixelech do skutečných rozměrů. Oblast zájmu je obdélníkový výřez vstupního obrazu. Obdélník je definován x , y souřadnicí levého horního rohu, dále svou šířkou a výškou. Souřadnice x je pozice levé bočnice. Souřadnice y je 130. řádek nad horním okrajem spinneru. Šířka oblasti odpovídá šířce spinneru a výška je 100 obrazových řádků. Stanovená oblast zájmu se v náhledu videa zobrazuje jako bílý obdélník. V případě že funkce selhává, je možné ji jednoduše vyřadit a definovat oblast ručně.

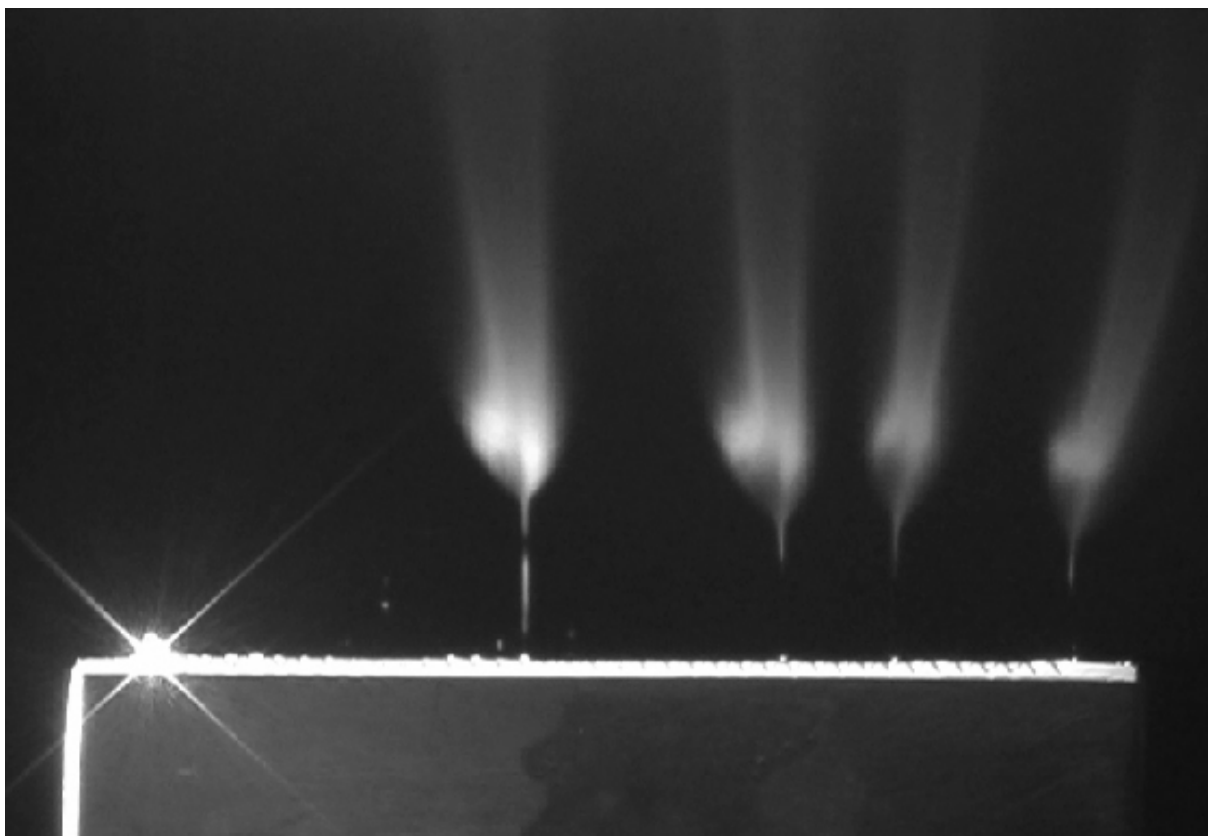
Přepočet rozměrů je stanoven podle rovnice (26):

$$l_r = \frac{r}{p} l_p, \quad (26)$$

kde je $l_r[mm]$ požadovaný rozměr, $r[mm]$ fyzický rozměr lišty, $p[px]$ rozměr lišty v obrazových bodech, $l_p[px]$ rozměr objektu v obrazových bodech.

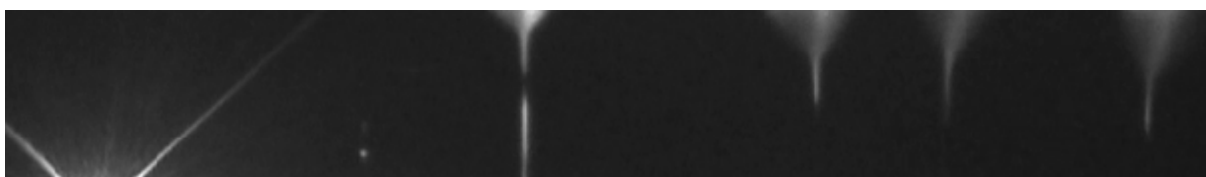
Registrace polohy trysek

Stanovení polohy trysek provádí funkce *AnalyzeIamge*, která implementuje metody analýzy obrazu popsané v kapitole 2.3. Vstupními parametry funkce jsou obraz, oblast zájmu, převodní konstanta, a dva strukturní elementy. Výstupy z funkce jsou těžiště trysek podle horizontální osy obrazu, rozestup trysek stanovený prostorovou analýzou obrazu a rozestup trysek stanovený frekvenční analýzou obrazu.



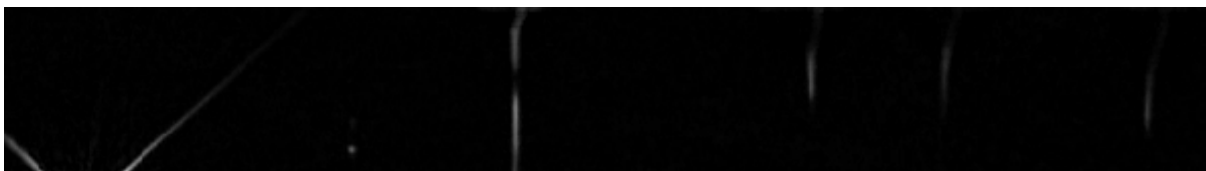
Obr. 27: *Vstupní obraz z videokamery. Zřetelné jsou 4 trysky, vlevo nečistota a nežádoucí světelné paprsky odražené od kapky, která se vytvořila na povrchu spinneru.*

Obraz je po vstupu do funkce ořezán na zajímavou část. Trysky se nenacházejí všechny ve stejné výšce, zajímavá oblast by měla zahrnovat i počátek oblasti rozšiřování trysek. Výřez je proveden funkcí *imcrop*, jejíž vstupní argumenty je celý obraz a koordináty oblasti zájmu.



Obr. 28: *Výřez zajímavé oblasti vytvořený bezprostředně po vstupu do funkce *AnalyzeImage*.*

Osvětlení scény nebývá stejnoměrné. Také rozšiřující se oblasti trysek by mohly vést k vzájemnému splynutí sousedních trysek, což by znemožnilo rozpoznání každé z nich. K odstranění nestejnoměrného jasů v celém obraze a zeštíhlení trysek je aplikována *top-hat* transformace, která je v Matlabu realizována pomocí funkce *imtophat*. Vstupní argumenty této funkce jsou výřez obrazu a *strukturní element*, kterým je v tomto případě čtverec o straně 7 pixelů. Výstup operace je vidět na Obr. 29.



Obr. 29: Výsledek *top-hat transformace* provedené nad Obr. 28, strukturním elementem je čtverec o straně 7px.

V obraze je vyrovnána nestejnóměrnost pozadí, šum je potlačen, zůstávají jen výrazně jasná místa. Důležité je odstranění rozšířených oblastí v horní části trysek. Stále v obraze zůstávají nežádoucí paprsky světla a nečistota v podobě 'ostrovu' v první třetině zleva. Výsledný obraz je dobrým vstupem pro *segmentaci*. Protože dané osvětlení scény způsobuje vyšší jas objektů v levé polovině obrazu, namísto *globální segmentace* je aplikována *segmentace lokální*. Obraz je horizontálně rozdělen na samostatné oblasti, které jsou nezávisle *segmentovány* metodou *prahování*. *Prahová hodnota* je při *lokální segmentaci* stanovena pro každou oblast zvlášť. Metoda *segmentace* pomocí *prahování* je v Matlabu implementovaná ve funkci *im2bw*. *Prahová hodnota* je stanovena pomocí funkce *graythresh*, která používá *Otsuovu metodu*. Jednotlivé oblasti jsou ukládány do *buněčného pole*, po *segmentaci* všech oblastí jsou *buňky* zpětně spojeny a *segmentovaný* obraz má shodné rozměry s obrazem vstupním. Pro správnou funkci dělení do oblastí je třeba zajistit, aby byla šířka zajímavé oblasti dělitelná počtem oblastí beze zbytku. V opačném případě je výsledný obraz na rozhraní oblastí deformován a šíře zpětně složeného obrazu neodpovídá rozměru obrazu na vstupu. Zvyšování počtu oblastí zlepšuje výsledky *segmentace*, zároveň se ale zvyšuje časová náročnost operace. V uvedeném příkladu je použito dělení obrazu uprostřed na dvě poloviny. Funkce *graythresh* nachází spolehlivě *prahovou hodnotu* v předloženém obraze tak, aby překrývání dvou množin v oblasti (*objekt, pozadí*), bylo minimální. To ovšem platí i v případě, že v předložené oblasti není přítomen ani jeden objekt – tryska. Pak je *prahová* hodnota nalezena mezi šumem a *segmentace* nevrací správný výsledek. Z tohoto důvodu je v algoritmu zavedena druhá *prahová* konstanta příslušející hodnotě šumu. V případě, že je práh z funkce *graythresh* roven, nebo je pod 2% jasového rozsahu, pak je pro *segmentaci* nastaven práh 100%, což způsobí zařazení všech obrazových bodů v oblasti do množiny '*pozadí*'. Výsledek *segmentace* je patrný z Obr. 30.



Obr. 30: *Výsledek lokální segmentace Obr. 29.*

Vstupní obraz je na základě jasových úrovní rozdělen do množin 'objekt' a 'pozadí'. Segmentací jsou zvýrazněny i nežádoucí prvky v obraze. Také nestejný odlesk na trysce uprostřed způsobil její rozdělení na dva různé objekty. Malé nespojitě částice lze odstranit pomocí *morfologického otevření*. K *morfologickému otevření* černobílého obrazu je v Matlabu určena funkce *bwareaopen*, pomocí které jsou z obrazu odstraněny objekty, jejichž plocha je menší než zadaný počet pixelů.



Obr. 31: *Morfologické otevření obrazu pomocí funkce `bwareaopen`. Použito bylo čtyřsousedství daného pixelu.*

Následkem *morfologického otevření* dojde k vyčištění obrazu od malých částic, které svou plochou zakrývají méně než 21 obrazových bodů. Tato hodnota byla stanovena na základě zkušenosti z provedených experimentů tak, aby byly vyřazeny nečistoty a zároveň zůstaly slabé trysky v obraze zachovány. Např. třetí objekt zleva, který byl v obraze přítomen až do předchozí operace, je nyní bezpečně eliminován. Problém nastal s tryskou uprostřed, která je rozdělena na dva různé objekty. K tomuto jevu dochází poměrně často a je zapříčiněn osvětlením scény. Tryska se leskne ve své spodní a horní části, zatímco uprostřed není patrná. Protože se v obraze snažíme nalézt objekty lineární a orientované svisle, je možné provést pokus o vyplnění mezery mezi objekty ležícími přímo nad sebou. V algoritmu jde o *morfologickou dilataci strukturním elementem* ve tvaru úsečky délky 10 pixelů, orientované v úhlu 90° od vodorovné osy obrazu. Operací dojde k protažení objektů o 20 pixelů ve vertikálním směru a objekty ležící přímo nad sebou se spojí v jeden. V Matlabu je *morfologická dilatace* realizována voláním funkce *imdilate*.



Obr. 32: Výsledek morfologické dilatace strukturní prvek je svislá úsečka délky 10px.

Všechny dosud provedené operace spadají svým charakterem do oblasti *předzpracování a segmentace* obrazu. Získaný obraz je zbaven šumu a nežádoucích artefaktů. Naopak rysy lineárních objektů jsou zvýrazněny. Dilatace je poslední operací nízké úrovně, kdy je vstupem i výstupem operace obraz původní velikosti. Další část algoritmu se věnuje *klasifikaci* objektů.

Jistý přechod od *segmentace* ke *klasifikaci* zprostředkovává funkce *bwlabel*. Tato funkce prochází černobílý obraz zleva doprava a označuje spojitě oblasti indexy tak, že prvky první nalezené spojitě oblasti (v černobílém obraze jde o prvky matice s hodnotou 1) jsou označeny číslem 1. Prvkům druhé spojitě oblasti je přiřazen index 2, atd. Výstupem funkce *bwlabel* je tzv. *label matrix*, jakási popisná matice, jejíž nenulové prvky označují svou hodnotou příslušnost k jedné spojitě oblasti v obraze.

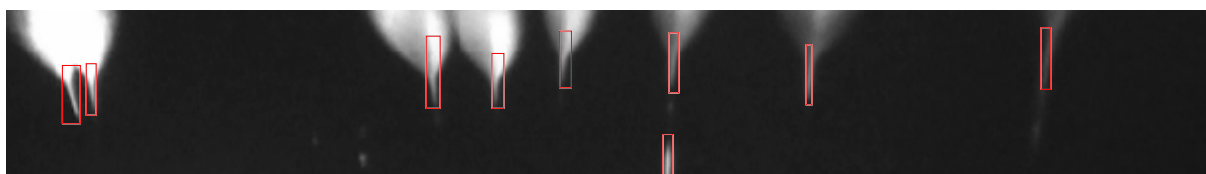


Obr. 33: Grafická reprezentace tzv. *label matrix* – matice označující spojitě oblasti indexy. V grafickém zobrazení *label matrix* je hodnota indexu příslušné oblasti vyjádřena jasovou úrovní. Objektu vlevo dole přísluší index 1, první objekt zprava je označen indexem 6.

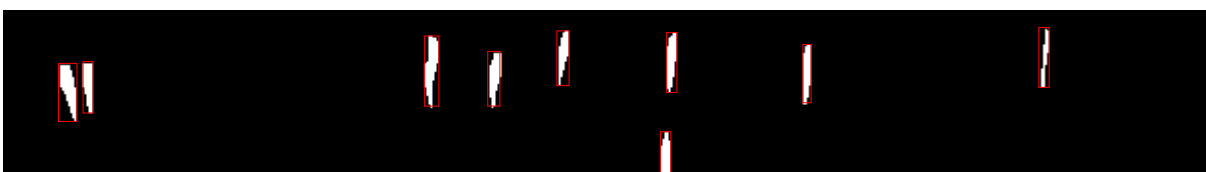
Popis charakteristických vlastností objektů v *popisné matici* realizuje funkce *regionprops*. Vstupní argumenty této funkce tvoří sada požadovaných *příznaků*, podle kterých je sestavena návratová struktura. K popisu objektů byly zvoleny tyto příznaky: *orientace*, *těžiště*, *opsaný obdélník*. Pro práci s *příznaky* objektů je nutné sestavit *klasifikační pravidla*. Aby mohl být objekt *klasifikován* jako tryska, musí splňovat tato dvě pravidla:

- orientace objektu musí být v intervalu 70° až 110° včetně; měřeno od vodorovné osy obrazu pootočením úhlu proti směru hodinových ručiček
- štíhlost objektu, definovaná jako poměr mezi výškou a šířkou opsaného obdélníku, musí být větší než 3.

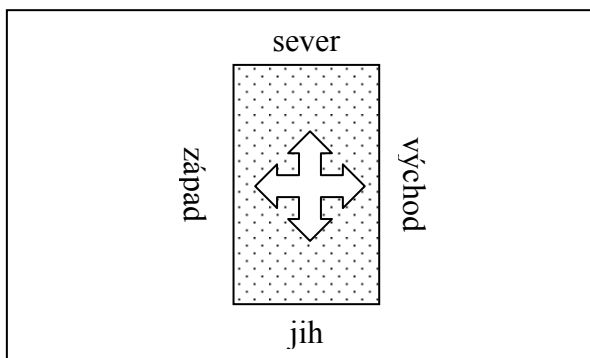
V prezentovaném příkladu jsou záměrně přítomny dva paprsky světla. Orientace těchto objektů je 120° a 55° , štíhlost 1.58 a 1.45. Ani jeden z těchto objektů nevyhovuje žádnému kritériu pro zařazení do třídy 'tryska', a proto jsou z množiny nalezených objektů vyřazeny. Data z provedených experimentů poukazovala na častou přítomnost objektů vzdálených od sebe pouhé jednotky pixelů. Následná diagnostika odhalila nedostatečné řešení problému s vertikálně 'roztrženou' tryskou. Jde o případ řešený pomocí *morfologické dilatace* na Obr. 32. Uspokojivé východisko přinesla až myšlenka rekonstrukce stavby 'podezřelých' objektů. Na Obr. 34 je znázorněna situace, kdy se přerušenu trysku (třetí zprava) nepodařilo *morfologickou dilatací* obnovit.



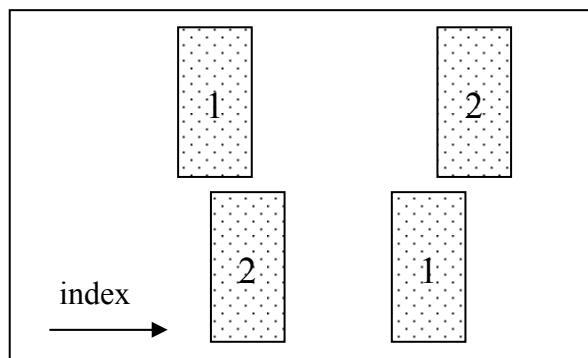
(a)



(b)



(c)



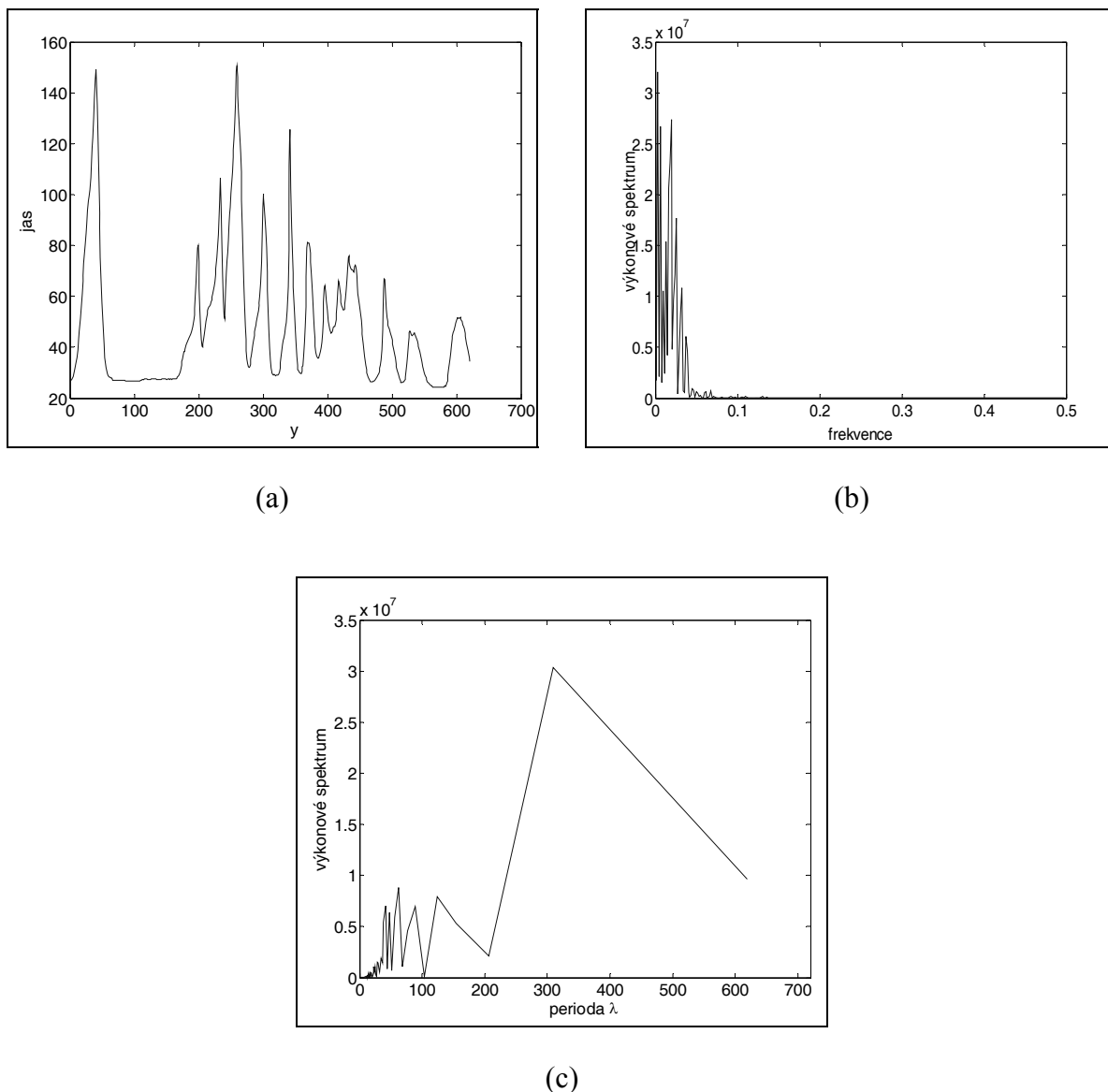
(d)

Obr. 34: Ilustrace problému s přerušenu tryskou; (a) originální obraz s detekovanými oblastmi trysek, (b) segmentovaný obraz s obdélníky opsanými kolem každého objektu. (c) konvence značení pro strany opsaného obdélníku užitá v těle funkce *reconstruct_jet*, (d) možné případy vzájemné polohy objektů s vyznačením indexů získaných pomocí *label matrix*. Z rozdílu mezi (a), (b) je také dobře patřené odstranění horního "vějíře" nad tryskami pomocí *top-hat* morfologie.

Přesto, že jde o jedinou trysku, byla v průběhu *segmentace* roztržena na dva nezávislé objekty *klasifikované* samostatně. Mohou nastat dvě situace tohoto typu, lišící se vzájemným postavením '*podezřelých*' trysek, jak je znázorněno na Obr. 34d. *Vnořená funkce*, která provádí rekonstrukci přerušovaných trysek (*reconstruct_jet*), je volána vždy po prvním průchodu množiny objektů *klasifikátorem*. Díky tomu můžeme na vstupu funkce považovat všechny prvky za objekty typu '*tryska*'. Po vstupu množiny objektů do funkce *reconstruct_jet* dojde k vytipování *podezřelých* trysek. Jde o takové, jejichž *západní a východní* stěny opsaného obdélníku se překrývají. Pokud je zároveň splněna podmínka, že se *severní a jižní* strana opsaného obdélníku nepřekrývají, pak jsou tyto dva podezřelé objekty považovány za trysku jedinou. Oba dva jsou z množiny vyjmuty a jsou nahrazeny tryskou jedinou, jejíž těžiště leží ve středu mezi původními dvěma tryskami. Pro zjištění vzdálenosti mezi tryskami v získané množině je třeba znát jen polohu jejich těžiště. Po prvním průchodu *klasifikátorem* a rekonstrukci trysek tak není důvod opsaný obdélník znovu dopočítávat.

Stanovením těžišť podle vodorovné osy obrazu pro všechny trysky tento postup registrace polohy končí. Vektor jednotlivých poloh je návratovou hodnotou funkce *AnalyzeImage*. Díky těmto údajům lze v průběhu experimentu, ale i po jeho skončení sledovat časový sled polohy každé trysky, její vznik a zánik v určitém místě, atd. Zpracováním získaných dat se zabývá samostatná kapitola této práce nazvaná Výsledky měření a jejich interpretace.

Podle teorie vyslovené týmem prof. Lukáše [1], by vznikající trysky měly být mezi sebou vzdáleny o vlnovou délku λ . To znamená, že jejich horizontální poloha na hladině polymeru má vykazovat jasnou periodicitu. K odhalení periodicity v signálu je vhodným nástrojem Fourierova analýza. Jednorozměrná Fourierova transformace a nalezení hlavní periody v obraze je ve funkci *AnalyzeImage* také implementováno. Vstupní signál představuje vektor získaný průměrováním jasových hodnot v jednotlivých sloupcích zajímavé oblasti. Jde tedy o průměr horizontálních profilů každého řádku, jak je znázorněno na Obr. 35. Přesto že nastávají situace, kdy obě metody podávají téměř shodný výsledek, neposkytuje frekvenční analýza obrazu zatím příliš spolehlivé výsledky. Rozhodně ji ale nelze vyloučit jako slepou cestu. Zřejmě by bylo potřeba lépe zpracovat vstupní data volbou jiné reprezentace obrazu, než poskytuje jeho průměrný jasový profil. Frekvenční analýza může být předmětem dalšího vývoje. Data získaná při experimentu tímto způsobem se zobrazují a ukládají. Pro statistické vyhodnocení je ale použita poloha trysek z jejich prostorového rozmístění, jehož popis tvoří hlavní část této kapitoly.



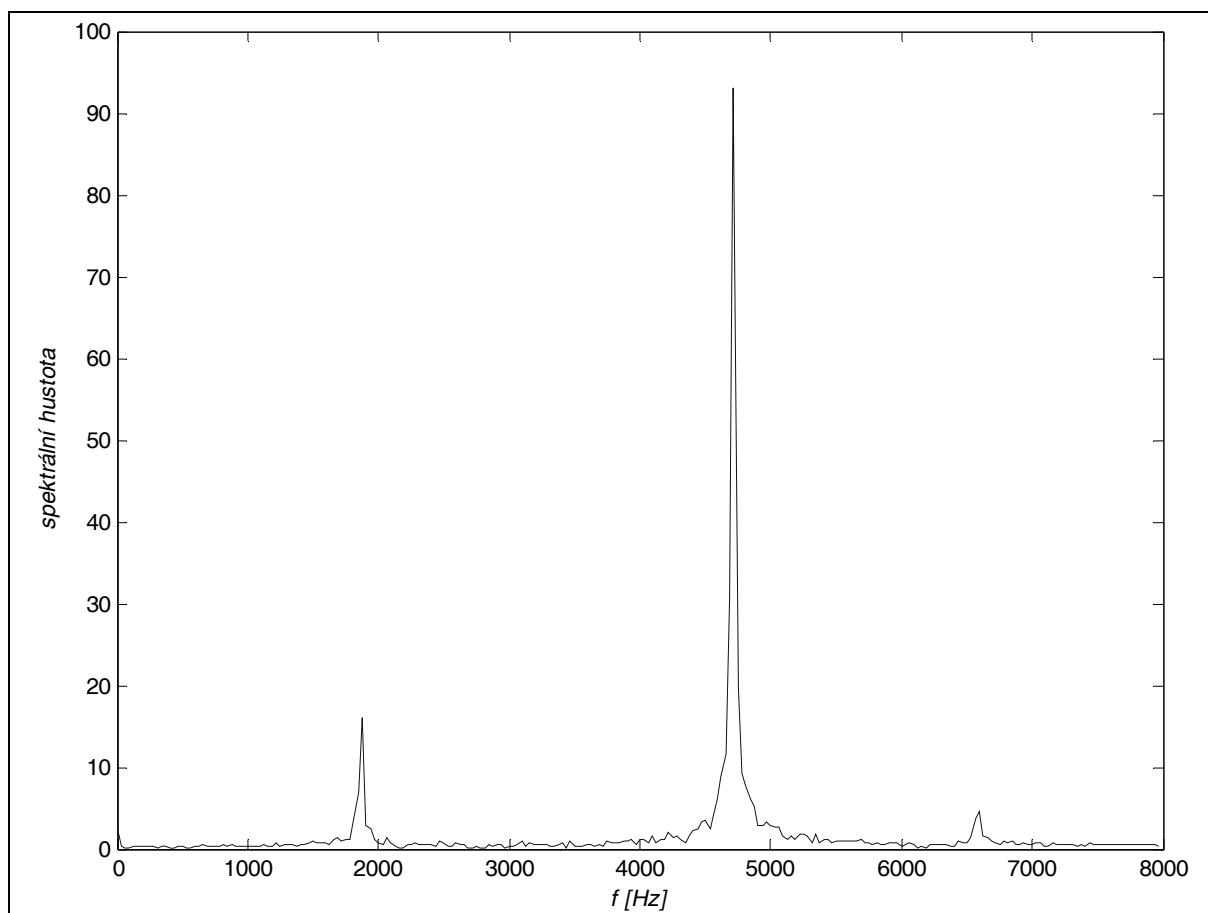
Obr. 35: Výsledky frekvenční analýzy pomocí Fourierovy transformace; (a) horizontální profil trysek, (b) frekvenční odezva, (c) periodogram.

Na konci funkce *AnalyzeImage* je implementováno jednoduché rozhodovací kritérium o správnosti určení zajímavé oblasti. V případě že konečná množina trysek obsahuje jen dva objekty a zároveň byly při *klasifikaci* vyřazeny více než 4 objekty, pak je určení oblasti trysek v obraze považováno za neplatné a před analýzou dalšího snímku je opětovně volána funkce *GetROI*. Aktuální snímek je tímto zneplatněn, avšak jeho opakovaná analýza s nově nalezenou oblastí se neprovádí. Důvodem k tomuto opatření je ohled na rychlost zpracování zejména při běhu v reálném čase. Protože je zpracováváno 25 snímků v jedné vteřině experimentu, o množství dat nouze není.

Registrace napětí zdroje

Pro zjištění aktuálního napětí elektrod slouží zvukový signál z převodníku. Při zpracování živého záznamu je díky *Data Acquisition Toolboxu* přístupováno na zvukovou kartu přes objekt analogového vstupu. Vstup zvukové karty je vzorkován s frekvencí 16kHz, a zpracováván postupem Fourierovy analýzy vždy po zachycení 4096 vzorků. Zpracování při živém vstupu je asynchronní a běží paralelně se zpracováním obrazu. Hodnota napětí dostupná aplikaci je tak aktualizována s frekvencí přibližně 4Hz. Při zpracování ze záznamu je vždy po načtení snímku nahrána do paměti příslušná délka zvukového vzorku následujícím způsobem. V paměti je $576 \times 720 \times N$ -rozměrná matice obrazových bodů, kdy N značí celkový počet snímků ve videu zachycených s frekvencí 25Hz, a $1 \times M$ -rozměrný vektor, který představuje zvukovou stopu. M je celkový počet vzorků: $M = F_s \cdot T$, kde F_s je vzorkovací frekvence ($F_s=16\text{kHz}$) a T je délka záznamu. Pak pro jeden snímek videa, který na časové ose představuje 1/25 vteřiny bylo při frekvenci 16kHz pořízeno $16000/25=640$ vzorků zvukové stopy. Při analýze obrazu ze záznamu je tedy pro každý snímek vybrán příslušný časový interval z vektoru zvukového záznamu a pomocí Fourierovy transformace je nalezena hlavní frekvence neboli tón. Před vstupem do algoritmu Rychlé Fourierovy transformace (*FFT*) je vzorek oříznut na velikost 512 prvků, to je 2^9 . Důvodem je způsob zpracování algoritmu *FFT*, k jehož vykonání je třeba vstupní vektor délky mocniny dvou.

Většina formátů pro ukládání multimediálních dat na PC dovoluje společné kódování videa i zvuku do jednoho souboru. *Kontejner*, který umožňuje kódování zvuku a videa v odlišných formátech do jednoho souboru, nese obecné pojmenování *AVI* (*Audio/Video Interleaving*). Matlab dokáže z jednoho souboru číst pouze jeden typ dat. K dekodování videa slouží funkce *aviread*, k dekodování audia je určena funkce *wavread*. Přičemž audio a videozáznam je třeba před vstupem do aplikace rozdělit na samostatné soubory. Při experimentech byl k účelům komprese videozáznamu a separaci audia od videa používán volně šiřitelný program *VirtualDub*. Podmínkou správného výpočtu napětí je audio stopa ve formátu *WAV* – PCM, 16kHz, 16bit, mono. Tomu odpovídá datový tok 256kb/s.



Obr. 36: Frekvenční spektrum signálu na výstupu převodníku při 29kV.

Kalibrace napětí

Po prvním spuštění aplikace je třeba kalibrovat převodník na napěťový zdroj. Kalibraci zprostředkovává funkce *AdjustVoltage*. Jde o nalezení statické charakteristiky – vztahu mezi napětím na vysokonapěťovém zdroji a příslušným tónem na výstupu převodníku. Charakteristika je lineární, lze ji popsat polynomem prvního stupně. Citlivost a posun nuly jsou stanoveny z proměřených bodů metodou nejmenších čtverců. Převodní funkce je ve tvaru: $U = kf + q$, kde k je citlivost převodníku, q posun nuly, U napětí na zdroji a f je výstupní frekvence na převodníku.

Poznámka: Veškeré funkce jsou volány z vnitřku aplikace na základě klikání počítačovou myší a jsou skryty za grafickým rozhraním. Uživatel nepřichází s funkcemi přímo do styku a provedení experimentu není ze strany náročnosti ovládání aplikace nijak komplikováno.

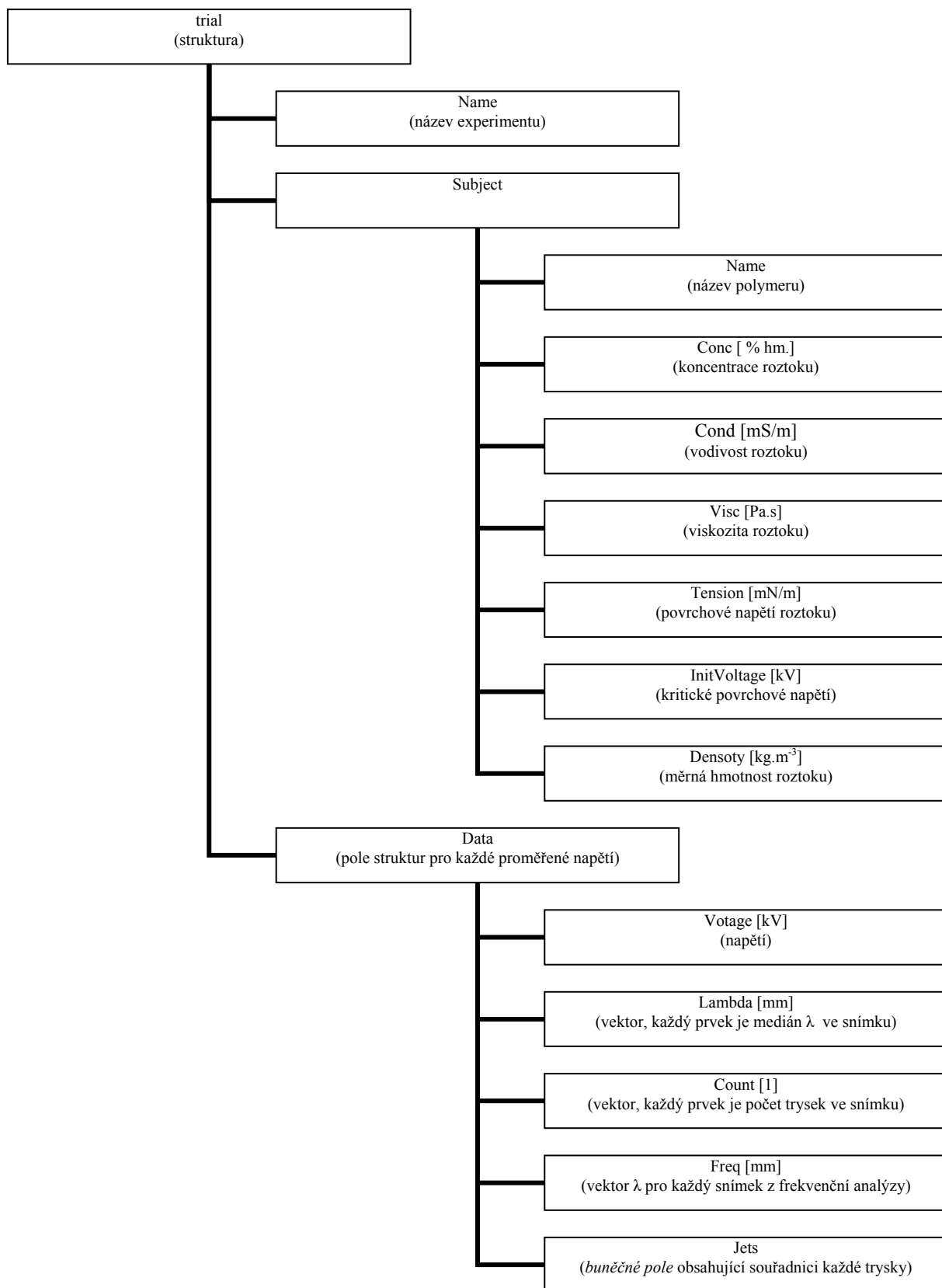
3.4 Výsledky měření a jejich interpretace

V pozadí aplikace jsou během experimentu polohy všech trysek ukládány do datové struktury, která kromě těchto údajů obsahuje také ostatní parametry experimentu. Tyto parametry jsou název experimentu, použitý polymer, koncentrace roztoku, viskozita roztoku, jeho hustota a vodivost. Dále povrchové napětí, kritické napětí, při kterém se začínají tvořit trysky a permitivita prostředí. Ve vnořené struktuře nazvané '*Data*' jsou pak ukládány hodnoty napětí a k němu příslušející mediánová hodnota rozestupů trysek v aktuálním snímku, počet trysek v aktuálním snímku, hlavní perioda mezi tryskami z frekvenční analýzy a *buněčné pole*, jehož každá *buňka* obsahuje vektor těžišť trysek v aktuálním snímku. Jednotky poloh a vzdáleností jsou v milimetrech. Poloha je vždy stanovena vzhledem k levému okraji celého vstupního obrazu. Graficky je datová struktura zobrazena na Obr. 37.

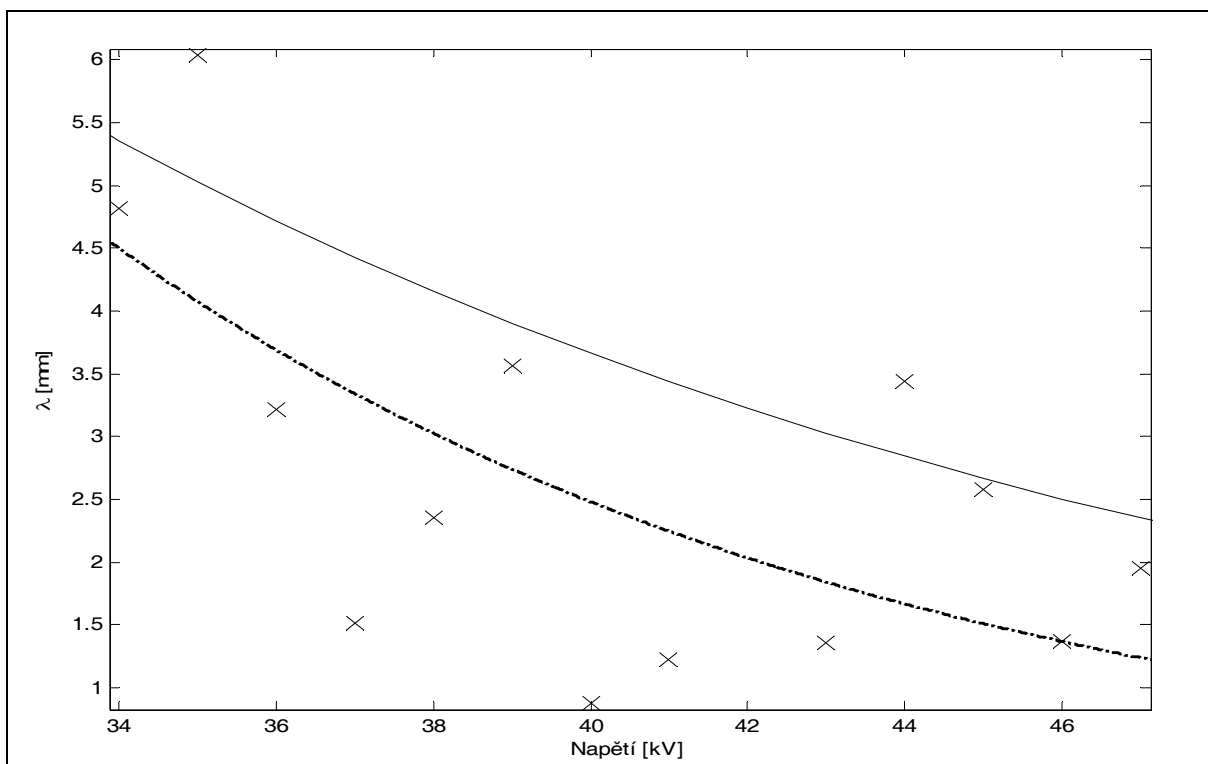
V průběhu zpracování videa je v aplikaci vykreslován aktuální snímek s vyznačenou zajímavou oblastí. Pod tryskami se objevují svislé bílé značky, které označují trysky nalezené algoritmem funkce *AnalyzeImage*. Tyto značky slouží jako průběžná vizuální kontrola úspěšnosti algoritmu. Pod tímto náhledem je umístěn rámec s textovými popisky. Zde jsou k nahlédnutí informace o dění v aktuálním snímku a průměrné hodnoty skrze snímky pro okamžité napětí. V pravém dolním rohu aplikace lze graficky sledovat časový průběh λ při aktuálním napětí na zdroji. Zobrazované hodnoty jsou odhady λ pomocí mediánu rozestupů trysek v aktuálním snímku. Graf se aktualizuje v intervalu 5 snímků. Při detekci změny napětí na zdroji dojde k vyčištění tohoto grafu a začínají se vykreslovat hodnoty pro nové napětí. Zároveň je vypočten průměr z časového průběhu λ , tedy ze spodního grafu, a hodnota je vynesena pro příslušné napětí do grafu v pravém horním rohu okna. Spodní graf ukazuje časovou variabilitu procesu a dobrým způsobem vyjadřuje odezvu trysek na změnu napětí. Horní graf vykresluje odhad λ ve vztahu k napětí elektrod. Již z tohoto grafu je zřejmá exponenciální závislost obou veličin.

Po skončení experimentu, je možné učinit komplexnější pohled na všechna získaná data. Aplikace dovoluje zobrazení tří základních závislostí.

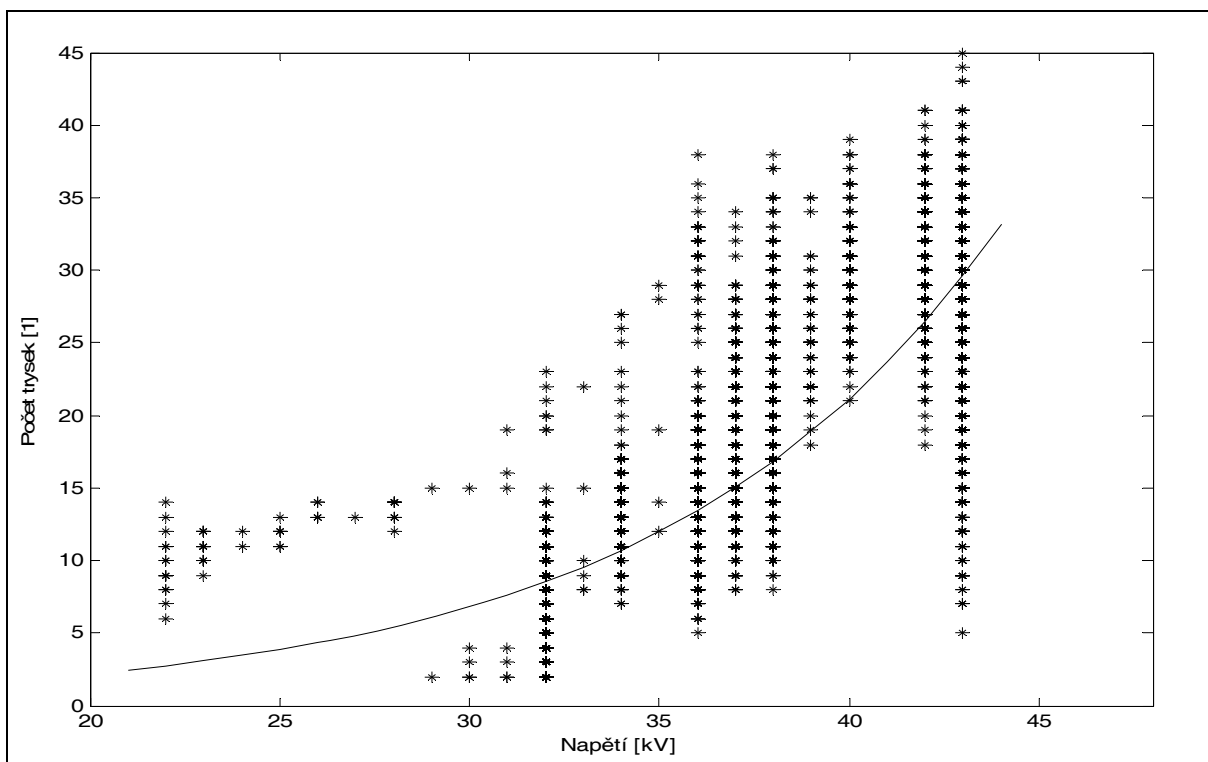
Rozestup trysek na napětí je zřejmý z Obr. 38 kde jednotlivé značky 'x' vyjadřují odhad λ jako nejčtenější rozestup trysek, který se při daném napětí na liště objevil. Přerušovaná čára znázorňuje proložení těchto dat exponenciálou. Smysl druhé spojité křivky vyjde najevo v dalším odstavci.



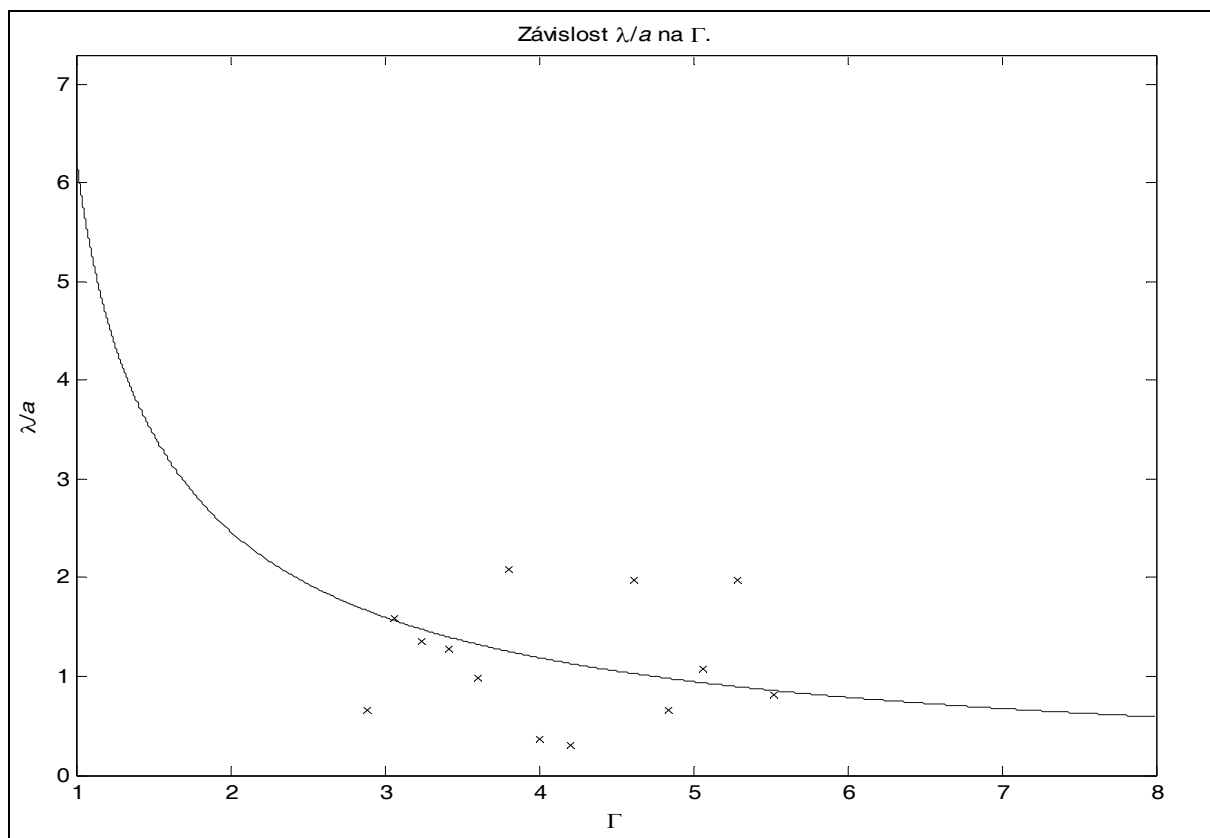
Obr. 37: Datová struktura pro ukládání informací o experimentu.



Obr. 38: Zobrazení vztahu rozestupu trysek a napětí elektrod.



Obr. 39: Vztah počtu trysek a napětí elektrod. Vodorovná osa vyjadřuje napětí, body na svislé ose jsou počty trysek, které byly v tomto napětí zjištěny.

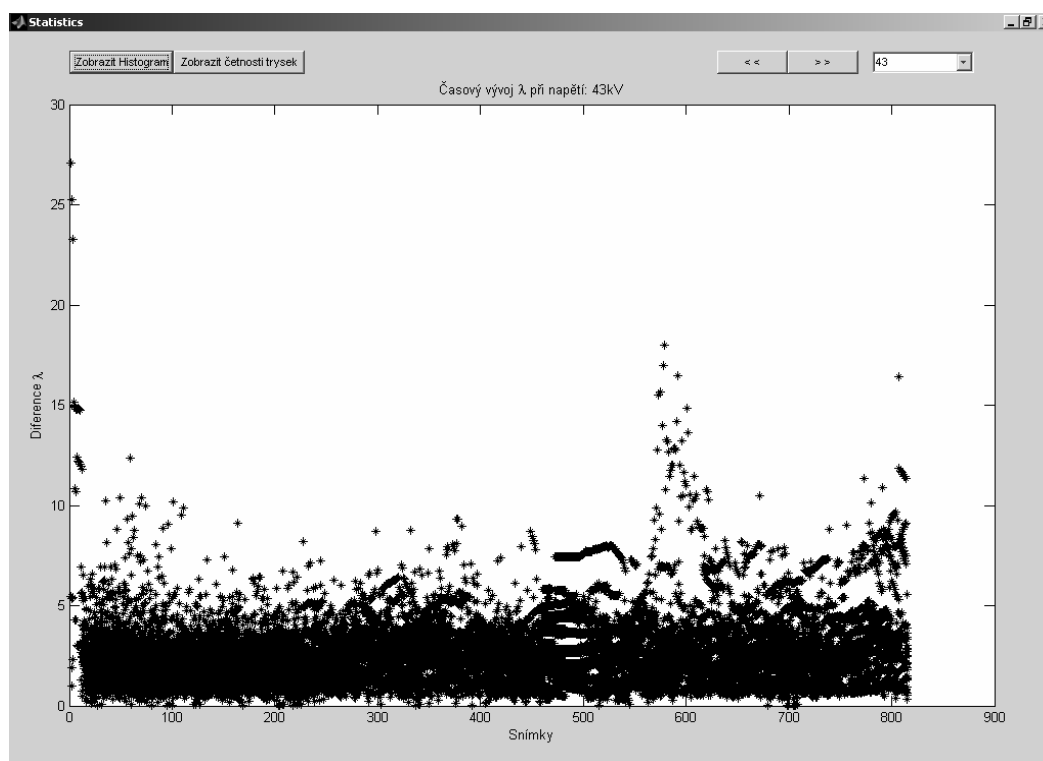


Obr. 40: *Závislost bezrozměrné vlnové délky λ/a na elektrospinningovém čísle Γ . Spojitá křivka představuje teoretickou závislost, kterou předurčuje teorie [1].*

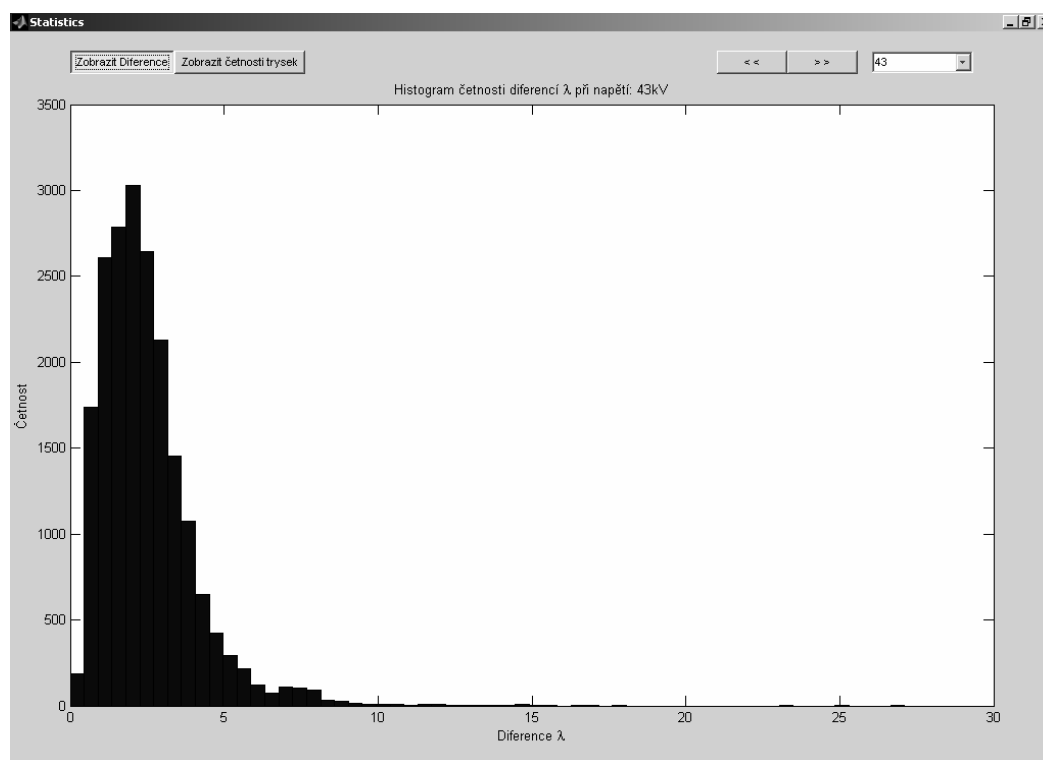
Počet trysek v závislosti na napětí vyjadřuje Obr. 39. Tento graf vynáší pro jednotlivá napětí všechny naměřené počty trysek detekované na hladině v průběhu pokusu. K proložení křivky byl použit zobecněný lineární model, kde $Y_i \sim \text{Poisson}(\mu_i)$, kde $\log(\mu_i) = \beta_1 + X_i\beta_2$. $\text{Poisson}(\mu_i)$ označuje Poissonovo rozdělení s parametrem μ . Jestliže počet trysek na liště známé délky exponenciálně roste, pak také vzájemné rozestupy mezi tryskami musí exponenciálně klesat. Pokud šířku zajímavé oblasti, ve které byly počty trysek stanoveny, podělíme proloženou křivkou, pak získáme další odhad vzdáleností jednotlivých trysek. Tento odhad je znázorněn na Obr. 38 jako spojitá křivka.

Vztah bezrozměrných veličin λ/a na Γ je vidět na Obr. 40.

Největší problém při zpracování dat představuje stanovení odhadu pro hodnotu rozestupu trysek λ . Pro náhled na všechny vzdálenosti, které se při konstantním napětí mezi tryskami objevily, slouží speciální diagram, uvedený na Obr. 41. Pomocí tohoto zobrazení lze listovat mezi výsledky měření podle napětí na zdroji a sledovat příslušné chování systému.



Obr. 41: Zobrazení všech zjištěných rozestupů mezi tryskami při konstantním napětí. Na vodorovné ose jsou vyneseny snímky zachycené při napětí 43kV. V každém snímku byly zjištěny vzájemné rozestupy trysek – ty jsou vyneseny na ose svislé jako jednotlivé body.



Obr. 42: Histogram odpovídající hodnotám z Obr. 41. Vodorovná osa zobrazuje rozestupy trysek, vertikální osa četnost daného rozměru.

Z viditelného tmavého pásu, tedy shluku hodnot ve spodní části grafu na Obr. 41, lze učinit odhad střední hodnoty λ . K jinému zobrazení distribuce vzdálenosti λ je možné vytvořit histogram z těchto dat, viz Obr. 42.

Rozdělení dat je zešíkmené. V některých případech, především pro nižší intenzity elektrického pole, má rozdělení více vrcholů. Z pohledu na histogram vyvstávají dvě otázky. Z jakého důvodu jsou v souboru zastoupeny téměř nulové a zároveň extrémně vysoké hodnoty λ ?

V kapitole *Registrace polohy trysek* je uvedeno řešení problému s nespojitým odleskem světla na trysce, který vede k tomu, že *klasifikátor* považuje jednu trysku za dva různé objekty s minimálním rozestupem. Přes uvedené řešení nastávají situace, kdy i rekonstrukce trysky selhává. V takových případech je zjištěná vzdálenost skutečně blízká nule. Také při vysokém napětí mohou na hladině v jednom bodě vznikat dvě trysky, které jsou z pohledu kamery za sebou. Díky ztrátě perspektivy se v obraze jeví jako pár trysek. Existuje také jiné, fyzikálně zajímavější vysvětlení. Čas od času totiž dojde k rozdělení trysky na dvě nové, přičemž jedna tryska začne na hladině putovat směrem od místa rozštěpení a druhá tryska zaniká. V tomto krátkém časovém intervalu jsou trysky skutečně ve vzájemně blízkém postavení.

Vysvětlení extrémních délek hledejme ve způsobu měření. K pokusům, které mají ověřit správnost teoretických předpokladů, byla použita kapilára ve tvaru kanálu. Nejedná se o žádné výrobní zařízení, jde čistě o pomůcku vyrobenou pro účely realizace pokusu. Teoretické vztahy ovšem nepostihují, ani nemohou, veškeré podmínky experimentu. Obecným předpokladem je zvláknování z rovnoměrné hladiny polymeru. Zajistit tuto podmínku je ale v praxi poměrně obtížný úkol. Na spinneru k rovnoměrnému rozlévání dávkované kapaliny po celé jeho šíři nedochází. Naopak lokálně vznikají na hladině kapky a zároveň jsou přítomna místa s nedostatečným množstvím roztoku. Geometrie aparatury navíc způsobuje, že k formování trysek těsně za kritickou hodnotou intenzity pole, dochází přednostně na okrajích lišty. Dávkování polymeru bylo při pokusech ovládáno manuálně pomocí injekční pumpy na základě pozorování stavu hladiny. Zpoždění vzniklé nedokonalým způsobem regulace bylo častou příčinou nedostatečného dávkování roztoku. Dochází pak k zániku trysek především uprostřed kapiláry. V této situaci je vzdálenost trysek rázem zcela mimo dosavadní průběh. Extrémně vybočující hodnoty jsou díky uvedeným příčinám vzniku považovány za irelevantní.

4 Závěr

Cílem této diplomové práce bylo sledování procesu elektrostatického zvlákňování z volné hladiny polymeru. Principu vzniku trysek na volné hladině polymeru je užito během výroby nanovláken z povrchu válce patentovanou technologií NanospiderTM. Pro popis samoorganizace hladiny do podoby trysek vlivem silného elektrostatického pole byla v současnosti vyslovena robustní teorie, pocházející z rukou týmu vědců Katedry netkaných textilií Technické Univerzity v Liberci, vedeného prof. Lukášem. K ověření teorie experimentem byla vybrána zjednodušená úloha, která spočívá v měření organizace trysek na jednorozměrné hladině polymeru. Za tímto účelem byl zkonstruován tzv. lištový spinner, který úzce svírá hladinu kapaliny mezi dvěma stěnami. Oblast hladiny a prostor nad ní byl během experimentů sledován videokamerou a následnou analýzou obrazu byly získány důležité informace o chování trysek v závislosti na aplikovaném napětí.

Za účelem analýzy obrazu a nalezení vhodného nástroje pro studium chování trysek nad volnou hladinou bylo nad výkonnými algoritmy naprogramováno též grafické uživatelské rozhraní. Jako vývojové prostředí byl zvolen systém Matlab společnosti MathWorks Inc. Vybudovaná aplikace umožňuje vyhodnocování experimentu v reálném čase ale i později z pořízeného videozáznamu. Potřebná zařízení představují díky celkové koncepci jen standardní PC se zvukovou kartou a jakákoli videokamera.

Získaná data přinesla zcela nový pohled na chování trysek během zvlákňování z volné hladiny polymeru v elektrostatickém poli. Vhodným zobrazením dat lze vypátrat např. časově závislé jevy, které zůstávají při okamžitém sledování procesu utajeny. Stanoveným cílem bylo odhalení závislosti počtu vznikajících trysek a jejich vzájemného rozestupu na různých intenzitách elektrického pole. Řešení tohoto úkolu vedlo k zaznamenávání polohy, zrodu a zániku každé trysky v průběhu experimentu. Rozšíření algoritmu získávání dat a nalezení jejich vhodného zpracování by mohly vést k odkrytí a pochopení dalších zákonitostí, podle kterých se hladina kapaliny řídí.

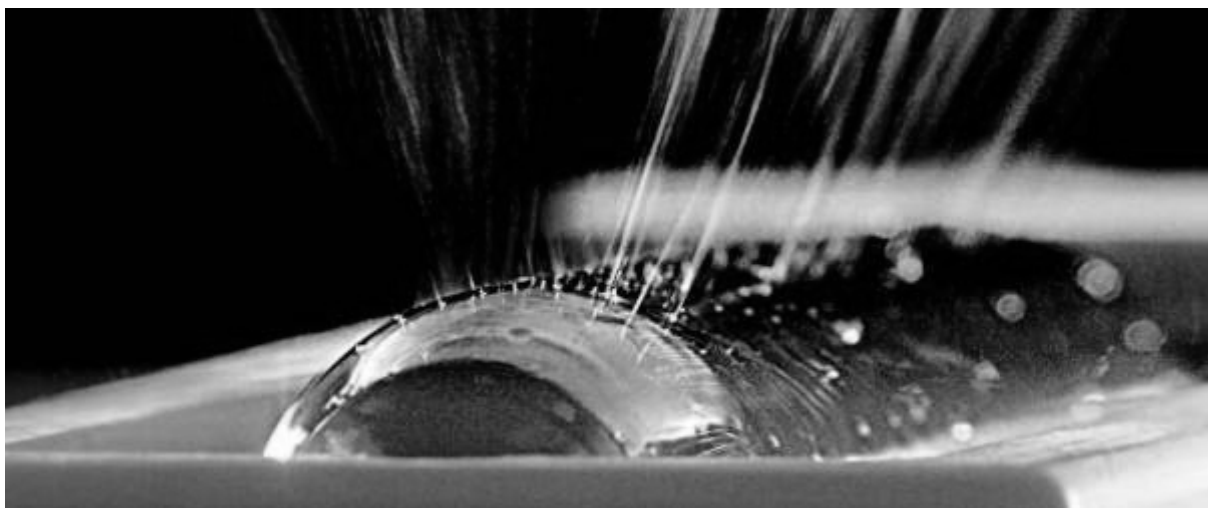
Díky výsledkům pokusů lze s jistotou konstatovat, že předpoklad exponenciální závislosti vzájemného rozestupu trysek na intenzitě elektrického pole byl potvrzen.

Jak již bylo řečeno, experimenty byly prováděny s kapilárou, která zaručuje podélný, téměř jednorozměrný tvar hladiny. Trysky díky tomu vznikají jen v jedné rovině a jejich polohu definují dvě souřadnice. Výška nad hladinou je navíc pro stanovení vzájemné vzdálenosti nepodstatnou informací, a tak k popisu polohy trysky stačí znát pozici na vodorovné ose. V

praxi se způsob výroby nanovláken na liště nepoužívá. Naopak, využívá se tvorby tenkého filmu na povrchu otáčejícího se válce. Na takto vytvořené dvourozměrné hladině dochází k vzniku trysek v kterémkoli místě. Jelikož se trysky objevují z čelního pohledu nejen vedle sebe, ale též za sebou, nelze jejich polohu stávající metodou řádně identifikovat. Uvažujme možnosti řešení tohoto problému pomocí dvou kolmo orientovaných kamer a pomocí kamery jedině.

Zachycením skutečné scény jedním obrazem ztrácíme perspektivu. Tedy informaci o vzdálenosti objektů od kamery podél osy snímání. Pro rekonstrukci scény lze použít principu stereoskopie, kdy je scéna popsána v jednom časovém okamžiku dvěma obrazy v odlišných pohledech. Pak spojením těchto dvou odpovídajících obrazů a ze známé polohy kamer je možné vypočíst hloubku scény.

Při fotografování zemského povrchu z velké výšky je třeba započítávat jeho zakřivení. Rozvinutí sférického povrchu Země do plochy, např. na stránku v atlasu, je zvládnutým postupem. Stejného principu by se pravděpodobně dalo využít pro snímání povrchu rotujícího válce. V této práci byl zvolen přístup sledování trysky nad hladinou kapaliny. Dosud nebyla zdůrazněna skutečnost, že rostoucí tryska je vlastně extrémní lokální zvýšení hladiny. Při vhodné poloze kamery a osvětlení je možné vznikající trysky pozorovat jako světlé body na jinak vyhlazeném povrchu válce. Příslušná ilustrace je na Obr. 43. Stačilo by tedy jednou kamerou snímat nikoli trysky, ale místa jejich původu. To znamená z textury válce a známé geometrie určit polohu trysek definovanou ve třech souřadnicích.



Obr. 43: Fotografie povrchu válce pořízená při výrobě vláken technologií NanospiderTM.

Analýza velmi nestabilního procesu elektrostatického zvlákňování byla v této práci uskutečněna na základě informací ze dvou senzorů. Těmi byla videokamera a voltmetr k získání informací o dění nad hladinou polymeru při definovaném napětí. V celém obvodu hrají ale důležitou roli ještě dva fyzikální parametry – množství polymeru v kapiláře a vzájemná poloha elektrod. Hodnoty těchto dvou parametrů nejsou při stávající analýze známe, přestože mají na proces zásadní vliv. Jistě by bylo v další práci dobré najít způsob jak oba dva parametry sledovat. Vzdálenost elektrod může být získána např. pomocí LVDT snímače polohy, nebo dalším využitím kamery a stanovením polohy horní elektrody z obrazu podobně jako je stanovena šířka spinneru. Jednoduchou informací o přeplnění kapiláry jsme schopni získat pomocí detekce kapky na hladině, která způsobuje odlesk světla. Tato detekce je v algoritmu hledání trysek vlastně již přítomna s tím, že zjištěné odlesky považujeme v obraze za nežádoucí a z množiny zajímavých objektů je vyřazujeme. Převodění informace o stavu procesu do digitální podoby je prvním krokem k sestavení plnohodnotného regulačního obvodu. Pokračování v činnosti započaté touto prací by mohlo v budoucnu vyústit v konstrukci bezobslužného průmyslového zařízení.

Užitá literatura:

- [1] Lukas D., Sarkar A., Pokorný P.: *Self organization of jets in electrospinning from free liquid surface – a general approach*, Journal of Applied Physics, 103, 8, 15. duben 2008.
- [2] Havlíček V, Pokorný M, Zemánek I.: *Elektrické obvody I*, Nakladatelství ČVUT, Praha, 2005.
- [3] Ramakrishna, S., Fujihara K., Teo W., Lim T., Ma, Z.: *An Introduction to Electrospinning and Nanofibers*, World Scientific Publishing, 2005.
- [4] Linfeld, G, Penny, J. *Numerical Methods Using Matlab*. [s.l.] : Ellis Horwood Limited, 1995. 323 s. ISBN 0-13-030966-4.
- [5] Ing. Václav Čížek, CSc.: *Diskrétní Fourierova transformace a její použití*; SNTL – Nakladatelství technické literatury, n. p., 1981; (04-019-81).
- [6] Doc. Ing. Miloš Sedláček, CSc.: *Zpracování signálů v měřicí technice*, Nakladatelství ČVUT, Praha, 1996.
- [7] Jirsák O., *Method of Nanofibres Production From A Polymer Solution Using Electrostatic Spinning And A Device For Carrying Out the Method*: Jirsák O., Sanetrník F., Lukáš D., Kotek V., Martinová L., Chaloupek J. Int.Cl.: H05B 7/00. United States Patent, 264/465; 264/468. Sep-8-2004.
- [8] Taylor G.; *Disintegration of Water Drops in an Electric Field*, Mathematical and Physical Sciences, 280, 1382, 1964.
- [9] Sobotka Z, Sobotka M: *Digitální zpracování a přenos obrazu II*, Dům techniky ČSVTS, Praha, 1987.
- [10] Wagner J., Kopal A.: *Fyzika, přehled pro strojní fakultu*, Vysoká škola strojní a textilní v Liberci, Liberec, 1985.
- [11] Zeleny, J: *Instability of Electrified Liquid Surfaces*, Physical Review, vol. 10, Issue 1, pp. 1-6.
- [12] Formhals A.: *Process and apparatus for preparing artificial threads*. US Patent 1,975,504, 1934.

Příloha 1: Instalace a použití aplikace

Příložené CD obsahuje adresář 'DP', ve kterém se nachází soubory MRCInstaller.exe, wmv9VCMsetup.exe, ReadMe.txt, CMP.ctf a CMP.exe. Tento adresář je třeba zkopírovat na místní disk hostitelského počítače. Soubor ReadMe.txt obsahuje některé údaje o verzi aplikace. Spuštěním souboru MRCInstaller (Matlab Component Runtime) obsahuje běhové prostředí a je třeba jej instalovat před spuštěním aplikace. Soubor wmv9VCMsetup.exe instaluje Windows Media Video Codec který používají přiložená videa. Po instalaci těchto dvou komponent je možné spustit aplikaci pomocí souboru CMP.exe. Adresář 'MMData' obsahuje vybrané video a audio soubory získané při experimentech. V adresáři 'Demo' je video, které zachycuje aplikaci během činnosti. Díky tomuto videu je možné sledovat činnost aplikace, aniž by bylo nutné ji ručně ovládat. Adresář 'Source' obsahuje kompletní zdrojový kód aplikace.

Po spuštění aplikace je třeba založit nový experiment **Soubor -> Nové měření...** vyplněním potřebných údajů. Poté jsou přístupné nabídky, které umožňují nastavit zdroj videa v nabídce **Zařízení -> Natavit zdroj....** Po připojení zdroje vysokého napětí je třeba provést kalibraci: **Zařízení -> Kalibrovat zdroj....** Při zpracování záznamu je nutné oddělit video od audia např. pomocí programu *VirtualDub* (<http://www.virtualdub.org>). V aplikaci je potřeba v nabídce **Soubor -> Import videa..., Import audia...** zvolit odpovídající datové proudy. Po spuštění aplikace je vždy zapnuta automatické nalezení hladiny polymeru. Tlačítkem **Analyzovat** se spustí vyhodnocování polohy trysek. Tlačítko **Zastavit** pracuje jako pauza. Pod náhledem videa je možné nalézt konkrétní pozici pomocí posuvníku. V případě že automatická detekce zajímavé oblasti neposkytuje dobré výsledky, lze tuto oblast definovat ručně kliknutím na přepínač **Definovat ROI**. Kurzor myši se změní na kříž. Zajímavá oblast je definována pomocí nákresu zajímavé oblasti v obraze myši. Po definování oblasti trysek je třeba napodruhé definovat obdélník kolem měřítka v obraze. Tím může být například celá lišta, ale také jakýkoli objekt v obraze o známé šířce. Šířku je pak nutné zadat do textového pole vlevo vedle ovládacích tlačítek. Zapnutí automatické detekce je provedeno opětovným stisknutím přepínače **Definovat ROI**. Po skončení analýzy obrazu je možné zjištěná data prohlížet pomocí diagramů a grafů v nabídce **Zobrazení**. Experiment a jeho vyhodnocení je možné ukládat pro další použití.

Příloha 2: Funkce *AnalyzeImage*

Funkce *AnalyzeImage* provádí vlastní zjištění polohy trysek v obraze. Reprezentuje nejdůležitější funkční prvek celé aplikace. Lze ji logicky rozdělit na části prostorové analýzy, frekvenční analýzy a zpětné vazby.

```
function [xCenters, lspat,lfreq, isROI] =...
    AnalyzeImage(frame, ROI, RULER, TopHatStrel, ImDilStrel)

% Prostorová analýza:
crop=imcrop(frame, ROI);
io=imtophat(crop, TopHatStrel);
d=2;region=cell(2,d);
h=size(crop,2);
for i=1:d
    region{1,i}=io(:, (i-1)*h/d+1:i*h/d);
    level=graythresh(region{1,i});
    level=level*(level>0.02);
    level=level+1*(level<=0.02);
    region{2,i}=im2bw(region{1,i}, level);% & (max(region{1,i}(:))>40);
end
bwin=[region{2,:}];
I=bwareaopen(bwin, 20, 4);
I=imdilate(I, ImDilStrel);
L=bwlabel(I);
stats = regionprops(L, 'Orientation', 'Centroid', 'BoundingBox');
stats(abs([stats.Orientation])<70)=[];
box=[stats.BoundingBox];
bwidth=box(3:4:end);bheight=box(4:4:end);
ind=bheight./bwidth<3;
stats(ind)=[];
box=[stats.BoundingBox];
xy=[stats.Centroid];
xs=xy(1:2:end);x=xs;
xs=reconstruct_jet;
function centers=reconstruct_jet
    west=box(5:4:end);
    east=box(1:4:end-5)+box(3:4:end-5);
    susp=find(east>west);
    jet=zeros(length(susp));
    if ~isempty(susp)
        for j=1:length(susp)
            box1=stats(susp(j)).BoundingBox;
            south1=box1(2)+box(4);
            north1=box1(2);
            box2=stats(susp(j)+1).BoundingBox;
            south2=box2(2)+box2(4);
            north2=box2(2);
            jet(j)=(south2<north1 | south1<north2).*susp(j);
        end
    end
    jet=jet(jet>0);
    for k=1:length(jet);
        x1=x(jet(k));
        x2=x(jet(k)+1);
        x(jet(k))=mean([x1 x2]);
        x(jet(k)+1)=0;
    end
end
```

Zbavení obrazu
nepodstatných částí.

Rozdělení obrazu,
lokální segmentace.

Zpětné složení obrazu.
Odstranění malých částic.
Spojení objektů nad sebou.
Popisná matice.
Popis příznaků, první klasifikace podle orientace.

Druhá klasifikace podle
štíhlosti.

Vnořená funkce pro rekonstrukci vadných trysek.

```

        centers=x(x>0);
    end
    fine=xs;
    % Frekvenční analýza:
    profile=mean(crop);
    T=ROI(3);
    Y=fft(double(profile));
    mag=abs(Y(1:round(T/2))).^2;
    df=1/T;
    f=0:df:(T/2-1)*df;
    p=1./f;
    mag(1:4)=0;
    [v,i]=max(mag);
    lfreq=NaN;
    if length(p)>=i
        lfreq=p(i)*RULER;
    end
    lspat=median(diff(sort(fine)))*RULER;
    xCenters=fine+ROI(1);

    %Kriterium nalezení správné oblasti:
    % ind je počet lineárních objektů
    % stats je počet objektů celkem
    if length(stats)>2 && length(ind)<5
        isROI=true;
    else
        isROI=false;
    end
end
end

```

Vektor vodorovných souřadnic trysek.

Vytvoření profilu obrazu.

Diskrétní Fourierova t.
Výkonové spektrum.

Vektor frekvencí.
Vektor period.

Hlavní perioda.

Převod z px->mm.

Odhad λ z mediánu.
Vektor pozic trysek.